SingularityCE User Guide

Release 4.0

SingularityCE Project Contributors

Nov 23, 2023

CONTENTS

1 Getting Started & Background Information **3**
- 1.1 What's New in SingularityCE 4.0 . . . 3
- 1.2 Introduction to SingularityCE . . . 4
- 1.3 Quick Start . . . 7
- 1.4 Security in SingularityCE . . . 22

2 Building Containers **25**
- 2.1 Build a Container . . . 25
- 2.2 Definition Files . . . 31
- 2.3 Build Environment . . . 51
- 2.4 Fakeroot feature . . . 56

3 Container Signing & Encryption **59**
- 3.1 Signing and Verifying Containers . . . 59
- 3.2 Key commands . . . 67
- 3.3 Encrypted Containers . . . 70

4 Sharing & Online Services **75**
- 4.1 Remote Endpoints . . . 75
- 4.2 Keyserver Management . . . 80
- 4.3 OCI Image Registries . . . 81
- 4.4 SCS Library . . . 82

5 Advanced Usage **91**
- 5.1 Bind Paths and Mounts . . . 91
- 5.2 Persistent Overlays . . . 99
- 5.3 Instances - Running Services . . . 104
- 5.4 Environment and Metadata . . . 113
- 5.5 Plugins . . . 126
- 5.6 Security Options . . . 129
- 5.7 Network virtualization . . . 134
- 5.8 Limiting Container Resources . . . 137

6 Compatibility **145**
- 6.1 Support for Docker and OCI Containers . . . 145
- 6.2 OCI Runtime Support . . . 167
- 6.3 SingularityCE and MPI applications . . . 177
- 6.4 GPU Support (NVIDIA CUDA & AMD ROCm) . . . 185

7 Get Involved **193**
- 7.1 Contributing . . . 193

8 Reference **197**
8.1 Appendix 197
8.2 Command Line Interface 214
8.3 Licenses 302

Welcome to the SingularityCE User Guide!

This guide aims to give an introduction to SingularityCE, brief installation instructions, and cover topics relevant to users building and running containers.

For a detailed guide to installation and configuration, as well as upgrading or uninstalling, please see the separate Admin Guide for this version of SingularityCE at https://sylabs.io/guides/4.0/admin-guide/.

CHAPTER

ONE

GETTING STARTED & BACKGROUND INFORMATION

1.1 What's New in SingularityCE 4.0

This section highlights important changes in SingularityCE 4.0 that are of note to users. See also the "What's New" section in the Admin Guide for administrator-facing changes.

1.1.1 OCI-mode

Singularity 4 introduces *OCI-mode* as a fully supported feature. It is enabled by using the `--oci` flag with the run / shell / exec / pull commands, or when `oci mode = yes` is set in `singularity.conf`.

In OCI-mode:

- Container images from OCI sources will be `pull`-ed to an *OCI-SIF* file. An OCI-SIF file encapsulates the OCI image configuration and squashed filesystem using an OCI, rather than Singularity specific, structure.
- The run / shell / exec commands use a low-level OCI runtime (crun/runc) for container execution.
- Default operation is *compatible with other OCI tools*, similar to `--compat` in Singularity's non-OCI native mode.
- OCI-mode supports running existing Singularity SIF images, that are not OCI-SIF, and can be made to *imitate native mode default behavior* by using the `--no-compat` flag.

1.1.2 CLI Changes

- The commands related to OCI/Docker registries that were under `singularity remote` have been moved to their own, dedicated *registry command*.
- The keyserver-related commands that were under `singularity remote` have been moved to their own, dedicated *keyserver command*.
- Adding a new remote endpoint using the `singularity remote add` command will now set the new endpoint as default. This behavior can be suppressed by supplying the `--no-default` (or `-n`) flag to `remote add`.
- `--cwd` is now the preferred form of the flag for setting the container's working directory, though `--pwd` is still supported for compatibility.

1.1.3 Runtime Behavior Changes

- The way `--home` is handled when running as root (e.g. `sudo singularity`) or with `--fakeroot` has changed. Previously, we were only modifying the `HOME` environment variable in these cases, while leaving the container's `/etc/passwd` file unchanged (with its homedir field pointing to `/root`, regardless of the value passed to `--home`). Now, both the value of HOME and the contents of `/etc/passwd` in the container will reflect the value passed to `--home`.
- Bind mounts are now performed in the order of their occurrence on the command line, or within the value of the `SINGULARITY_BIND` environment variable. (Previously, image-mounts were always performed first, regardless of order.)
- The current working directory is created in the container when it doesn't exist, so that it can be entered. You must now specify `--no-mount home,cwd` instead of just `--no-mount home` to avoid mounting from `$HOME` if you run `singularity` from inside `$HOME`.
- If the path of the current working directory in the container and on the host contain symlinks to different locations, the current working directory will not be mounted.

1.1.4 New Features & Functionality

- Templating support for definition files: users can now *define variables in definition files* via a matching pair of double curly brackets.
- Added a `--secret` flag (shorthand: `-s`) to the `key remove` subcommand, to allow removal of a private key by fingerprint.
- Added `--private` as a synonym for `--secret` in `key list`, `key export`, and `key remove` subcommands.
- The `instance start` command now accepts an optional `--app <name>` argument which invokes the start script within the `%appstart <name>` section in the definition file. The `instance stop` command still only requires the instance name.
- A new `--no-pid` flag for `singularity run/shell/exec` disables the PID namespace inferred by `--containall` and `--compat`.
- *Caching of OCI images* is now architecture aware. This fixes behavior in cases where a user's home directory is shared between systems of different architectures. If you do not use older versions of SingularityCE on a system, you can remove obsolete cache entries with `singularity cache clean --type blob`.
- A new `--platform` flag can be used to specify an `OS/Architecture[/Variant]` when *pulling images from OCI* or *library* sources. When pulling from library sources the optional variant is ignored.
- The `--arch` flag can now be used to specify a required architecture when *pulling images from OCI*, as well as *library* sources.

1.2 Introduction to SingularityCE

SingularityCE is a *container* platform. It allows you to create and run containers that package up pieces of software in a way that is portable and reproducible. You can build a container using SingularityCE on your laptop, and then run it on many of the largest HPC clusters in the world, local university or company clusters, a single server, in the cloud, or on a workstation down the hall. Your container is a single file, and you don't have to worry about how to install all the software you need on each different operating system.

1.2.1 Why use SingularityCE?

SingularityCE was created to run complex applications on HPC clusters in a simple, portable, and reproducible way. First developed at Lawrence Berkeley National Laboratory, it quickly became popular at other HPC sites, academic sites, and beyond. SingularityCE is an open-source project, with a friendly community of developers and users. The user base continues to expand, with SingularityCE now used across industry and academia in many areas of work.

Many container platforms are available, but SingularityCE is focused on:

- Verifiable reproducibility and security, using cryptographic signatures, an immutable container image format, and in-memory decryption.
- Integration over isolation by default. Easily make use of GPUs, high speed networks, parallel filesystems on a cluster or server by default.
- Mobility of compute. The single file SIF container format is easy to transport and share.
- A simple, effective security model. You are the same user inside a container as outside, and cannot gain additional privilege on the host system by default. Read more about *Security in SingularityCE*.

1.2.2 Why use containers?

A Unix operating system is broken into two primary components, the kernel space, and the user space. The Kernel talks to the hardware, and provides core system features. The user space is the environment that most people are most familiar with. It is where applications, libraries and system services run.

Traditionally you use an operating system that has a fixed combination of kernel and user space. If you have access to a machine running CentOS then you cannot install software that was packaged for Ubuntu on it, because the user space of these distributions is not compatible. It can also be very difficult to install multiple versions of the same software, which might be needed to support reproducibility in different workflows over time.

Containers change the user space into a swappable component. This means that the entire user space portion of a Linux operating system, including programs, custom configurations, and environment can be independent of whether your system is running CentOS, Fedora etc., underneath. A SingularityCE container packages up whatever you need into a single, verifiable file.

Software developers can now build their stack onto whatever operating system base fits their needs best, and create distributable runtime environments so that users never have to worry about dependencies and requirements, that they might not be able to satisfy on their systems.

1.2.3 Use Cases

BYOE: Bring Your Own Environment!

Engineering work-flows for research computing can be a complicated and iterative process, and even more so on a shared and somewhat inflexible production environment. SingularityCE solves this problem by making the environment flexible.

Additionally, it is common (especially in education) for schools to provide a standardized pre-configured Linux distribution to the students which includes all of the necessary tools, programs, and configurations so they can immediately follow along.

Reproducible science

SingularityCE containers can be built to include all of the programs, libraries, data and scripts such that an entire demonstration can be contained and either archived or distributed for others to replicate no matter what version of Linux they are presently running.

Commercially supported code requiring a particular environment

Some commercial applications are only certified to run on particular versions of Linux. If that application was installed into a SingularityCE container running the version of Linux that it is certified for, that container could run on any Linux host. The application environment, libraries, and certified stack would all continue to run exactly as it is intended.

Additionally, SingularityCE blurs the line between container and host such that your home directory (and other directories) exist within the container. Applications within the container have full and direct access to all files you own thus you can easily incorporate the contained commercial application into your work and process flow on the host.

Static environments (software appliances)

Fund once, update never software development model. While this is not ideal, it is a common scenario for research funding. A certain amount of money is granted for initial development, and once that has been done the interns, grad students, post-docs, or developers are reassigned to other projects. This leaves the software stack un-maintained, and even rebuilds for updated compilers or Linux distributions can not be done without unfunded effort.

Legacy code on old operating systems

Similar to the above example, while this is less than ideal it is a fact of the research ecosystem. As an example, I know of one Linux distribution which has been end of life for 15 years which is still in production due to the software stack which is custom built for this environment. SingularityCE has no problem running that operating system and application stack on a current operating system and hardware.

Complicated software stacks that are very host specific

There are various software packages which are so complicated that it takes much effort in order to port, update and qualify to new operating systems or compilers. The atmospheric and weather applications are a good example of this. Porting them to a contained operating system will prolong the usefullness of the development effort considerably.

Complicated work-flows that require custom installation and/or data

Consolidating a work-flow into a SingularityCE container simplifies distribution and replication of scientific results. Making containers available along with published work enables other scientists to build upon (and verify) previous scientific work.

1.3 Quick Start

This guide is intended for running SingularityCE on a computer where you have root (administrative) privileges, and will install SingularityCE from source code. Other installation options, including building an RPM package and installing SingularityCE without root privileges, are discussed in the installation section of the admin guide.

If you need to request an installation on your shared resource, see the section on *requesting an installation* for information to send to your system administrator.

For any additional help or support contact the Sylabs team: https://www.sylabs.io/contact/

1.3.1 Quick Installation Steps

You will need a Linux system to run SingularityCE natively. Options for using SingularityCE on Mac and Windows machines, along with alternate Linux installation options, are discussed in the installation section of the admin guide.

If you have an existing version of SingularityCE installed from source, which you wish to upgrade or remove / uninstall, see the installation section of the admin guide.

Prerequisites

Install system dependencies

You must first install development tools and libraries to your host.

On Debian-based systems, including Ubuntu:

```
# Ensure repositories are up-to-date
sudo apt-get update
# Install debian packages for dependencies
sudo apt-get install -y \
   autoconf \
   automake \
   cryptsetup \
   git \
   libfuse-dev \
   libglib2.0-dev \
   libseccomp-dev \
   libtool \
   pkg-config \
   runc \
   squashfs-tools \
   squashfs-tools-ng \
   uidmap \
   wget \
   zlib1g-dev
```

On versions 8 or later of RHEL / Alma Linux / Rocky Linux, as well as on Fedora:

```
# Install basic tools for compiling
sudo yum groupinstall -y 'Development Tools'
# Install RPM packages for dependencies
sudo yum install -y \
```

(continues on next page)

(continued from previous page)

```
   autoconf \
   automake \
   crun \
   cryptsetup \
   fuse3-devel \
   git \
   glib2-devel \
   libseccomp-devel \
   libtool \
   squashfs-tools \
   wget \
   zlib-devel
```

On version 7 of RHEL / CentOS:

```
# Install basic tools for compiling
sudo yum groupinstall -y 'Development Tools'
# Install RPM packages for dependencies
sudo yum install -y \
   autoconf \
   automake \
   cryptsetup \
   fuse3-devel \
   git \
   glib2-devel \
   libseccomp-devel \
   libtool \
   runc \
   squashfs-tools \
   wget \
   zlib-devel
```

On SLES / openSUSE Leap:

```
# Install RPM packages for dependencies
sudo zypper in \
 autoconf \
 automake \
 cryptsetup \
 fuse3-devel \
 gcc \
 gcc-c++ \
 git \
 glib2-devel \
 libseccomp-devel \
 libtool \
 make \
 pkg-config \
 runc \
 squashfs \
 wget \
 zlib-devel
```

Install sqfstar / tar2sqfs for OCI-mode

If you intend to use the *OCI mode* of SingularityCE, your system must provide either:

- `squashfs-tools` / `squashfs` >= 4.5, which provides the `sqfstar` utility. Note that older versions of these packages, provided by many distributions, do not include `sqfstar`.
- `squashfs-tools-ng`, which provides the `tar2sqfs` utility. This is not packaged by all distributions.

Below are instructions on how to obtain one of these two utilities on various distributions.

Debian / Ubuntu

On Debian/Ubuntu `squashfs-tools-ng` is available in the distribution repositories. It has been included in the *Install system dependencies* step above. No further action is necessary.

Fedora

On Fedora, the `squashfs-tools` package, included in the *Install system dependencies* step above, includes *sqfstar*. No further action is necessary.

RHEL / Alma Linux / Rocky Linux / CentOS

On RHEL and derivatives, a COPR is available at: https://copr.fedorainfracloud.org/coprs/dctrud/squashfs-tools-ng/

This COPR provides `squashfs-tools-ng`, which will not replace any standard EL or EPEL packages. To use it:

EL 8 / 9

```
sudo dnf install dnf-plugins-core
sudo dnf copr enable dctrud/squashfs-tools-ng
sudo dnf install squashfs-tools-ng
```

EL 7

```
sudo yum install yum-plugin-copr
sudo yum copr enable dctrud/squashfs-tools-ng
sudo yum install squashfs-tools-ng
```

SLES / openSUSE Leap

On SLES/openSUSE, follow the instructions at the filesystems project to obtain a more recent `squashfs` package, which provides `sqfstar`.

Next steps

You are now ready to install SingularityCE. There are 3 broad steps to installing SingularityCE itself:

1. *Installing Go*
2. *Downloading SingularityCE*
3. *Compiling SingularityCE Source Code*

Install Go

SingularityCE is written in Go, and may require a newer version of Go than is available in the repositories of your distribution. We recommend installing the latest version of Go from the official binaries.

SingularityCE aims to maintain support for the two most recent stable versions of Go. This corresponds to the Go Release Maintenance Policy and Security Policy, ensuring critical bug fixes and security patches are available for all supported language versions.

Note: If you have previously installed Go from a download, rather than an operating system package, it is important that you remove your `go` directory, e.g. `rm -r /usr/local/go`, before installing a newer version. Extracting a new version of Go over an existing installation can lead to errors when building Go programs, as it may leave behind old files, which have been removed or replaced in newer versions.

Visit the Go Downloads page and pick a package archive suitable to the environment you are in. Once the download is complete, extract the archive to `/usr/local` (or follow other instructions on the Go installation page). Alternatively, follow the commands here, making sure to replace specific values as needed:

```
$ export VERSION=1.21.0 OS=linux ARCH=amd64 && \
  wget https://dl.google.com/go/go$VERSION.$OS-$ARCH.tar.gz && \
  sudo tar -C /usr/local -xzvf go$VERSION.$OS-$ARCH.tar.gz && \
  rm go$VERSION.$OS-$ARCH.tar.gz
```

Set the Environment variable `PATH` to point to Go:

```
$ echo 'export PATH=/usr/local/go/bin:$PATH' >> ~/.bashrc && \
  source ~/.bashrc
```

Download SingularityCE from a release

You can download SingularityCE from one of the releases. To see a full list, visit the GitHub release page. After deciding on a release to install, you can run the following commands to proceed with the installation.

```
$ export VERSION=4.0.0 && \
    wget https://github.com/sylabs/singularity/releases/download/v${VERSION}/singularity-
↪ce-${VERSION}.tar.gz && \
    tar -xzf singularity-ce-${VERSION}.tar.gz && \
    cd singularity-ce-${VERSION}
```

Note: Do not attempt to build SingularityCE from the *Source code (zip)* or *Source code (tar.gz)* archives that are auto-generated by GitHub. These do not include some of the required internal dependencies needed to build SingularityCE. Instead, to build SingularityCE from source, use the archive named **singularity-ce-${VERSION}.tar.gz**.

Compile the SingularityCE source code

Now you are ready to build SingularityCE. Internal dependencies will be automatically downloaded. You can build SingularityCE using the following commands:

```
$ ./mconfig && \
    make -C builddir && \
    sudo make -C builddir install
```

Note: SingularityCE must be installed as root to function properly.

1.3.2 Overview of the SingularityCE Interface

SingularityCE's *command line interface* allows you to build and interact with containers transparently. You can run programs inside a container as if they were running on your host system. You can easily redirect I/O, use pipes, pass arguments, and access files, sockets, and ports on the host system from within a container.

The `help` command gives an overview of SingularityCE options and subcommands as follows:

```
$ singularity help

Linux container platform optimized for High Performance Computing (HPC) and
Enterprise Performance Computing (EPC)

Usage:
  singularity [global options...]

Description:
  Singularity containers provide an application virtualization layer enabling
  mobility of compute via both application and environment portability. With
  Singularity one is capable of building a root file system that runs on any
  other Linux system where Singularity is installed.

Options:
```

(continues on next page)

(continued from previous page)

```
  -c, --config string   specify a configuration file (for root or
                        unprivileged installation only) (default
                        "/usr/local/etc/singularity/singularity.conf")
  -d, --debug           print debugging information (highest verbosity)
  -h, --help            help for singularity
      --nocolor         print without color output (default False)
  -q, --quiet           suppress normal output
  -s, --silent          only print errors
  -v, --verbose         print additional information
      --version         version for singularity

Available Commands:
  build       Build a Singularity image
  cache       Manage the local cache
  capability  Manage Linux capabilities for users and groups
  completion  Generate the autocompletion script for the specified shell
  config      Manage various singularity configuration (root user only)
  delete      Deletes requested image from the library
  exec        Run a command within a container
  help        Help about any command
  inspect     Show metadata for an image
  instance    Manage containers running as services
  key         Manage OpenPGP keys
  keyserver   Manage singularity keyservers
  oci         Manage OCI containers
  overlay     Manage an EXT3 writable overlay image
  plugin      Manage Singularity plugins
  pull        Pull an image from a URI
  push        Upload image to the provided URI
  registry    Manage authentication to OCI/Docker registries
  remote      Manage singularity remote endpoints
  run         Run the user-defined default command within a container
  run-help    Show the user-defined help for an image
  search      Search a Container Library for images
  shell       Run a shell within a container
  sif         Manipulate Singularity Image Format (SIF) images
  sign        Add digital signature(s) to an image
  test        Run the user-defined tests within a container
  verify      Verify digital signature(s) within an image
  version     Show the version for Singularity

Examples:
  $ singularity help <command> [<subcommand>]
  $ singularity help build
  $ singularity help instance start


For additional help or support, please visit https://www.sylabs.io/docs/
```

Information about individual subcommands can also be viewed by using the `help` command:

```
$ singularity help verify
```

(continues on next page)

(continued from previous page)

```
Verify digital signature(s) within an image

Usage:
  singularity verify [verify options...] <image path>

Description:
  The verify command allows a user to verify one or more digital signatures
  within a SIF image.

  Key material can be provided via PEM-encoded file, or via the PGP keyring. To
  manage the PGP keyring, see 'singularity help key'.

Options:
  -a, --all                               verify all objects
      --certificate string                path to the certificate
      --certificate-intermediates string  path to pool of intermediate
                                          certificates
      --certificate-roots string          path to pool of root certificates
  -g, --group-id uint32                   verify objects with the
                                          specified group ID
  -h, --help                              help for verify
  -j, --json                              output json
      --key string                        path to the public key file
      --legacy-insecure                   enable verification of
                                          (insecure) legacy signatures
  -l, --local                             only verify with local key(s)
                                          in keyring
      --ocsp-verify                       enable online revocation check
                                          for certificates
  -i, --sif-id uint32                     verify object with the specified ID
  -u, --url string                        specify a URL for a key server


Examples:
  Verify with a public key:
  $ singularity verify --key public.pem container.sif

  Verify with PGP:
  $ singularity verify container.sif


For additional help or support, please visit https://www.sylabs.io/docs/
```

SingularityCE uses positional syntax (i.e., the order of commands and options matters). Global options affecting the behavior of all commands follow immediately after the main `singularity` command. Then come subcommands, followed by their options and arguments.

For example, to pass the `--debug` option to the main `singularity` command and run SingularityCE with debugging messages on:

```
$ singularity --debug run library://lolcow
```

To pass the `--containall` option to the `run` command and run a SingularityCE image in an isolated manner:

```
$ singularity run --containall library://lolcow
```

SingularityCE 2.4 introduced the concept of command groups. For instance, to list Linux capabilities for a particular user, you would use the `list` command in the `capability` command group, as follows:

```
$ singularity capability list myuser
```

Container authors might also write help docs specific to a container, or for an internal module called an "app". If those help docs exist for a particular container, you can view them as follows:

```
$ singularity inspect --helpfile container.sif  # See the container's help, if provided

$ singularity inspect --helpfile --app=foo foo.sif  # See the help for the app foo, if␣
↪provided
```

1.3.3 Download pre-built images

You can use the `search` command to locate groups, collections, and containers of interest on the Container Library .

```
$ singularity search tensorflow
Found 22 container images for amd64 matching "tensorflow":

    library://ajgreen/default/tensorflow2-gpu-py3-r-jupyter:latest
            Current software: tensorflow2; py3.7; r; jupyterlab1.2.6
            Signed by: 1B8565093D80FA393BC2BD73EA4711C01D881FCB

    library://bensonyang/collection/tensorflow-rdma_v4.sif:latest

    library://dxtr/default/hpc-tensorflow:0.1

    library://emmeff/tensorflow/tensorflow:latest

    library://husi253/default/tensorflow:20.01-tf1-py3-mrcnn-2020.10.07

    library://husi253/default/tensorflow:20.01-tf1-py3-mrcnn-20201014

    library://husi253/default/tensorflow:20.01-tf2-py3-lhx-20201007

    library://irinaespejo/default/tensorflow-gan:sha256.
↪0c1b6026ba2d6989242f418835d76cd02fc4cfc8115682986395a71ef015af18

    library://jon/default/tensorflow:1.12-gpu
            Signed by: D0E30822F7F4B229B1454388597B8AFA69C8EE9F

    ...
```

You can use the *pull* and *build* commands to download pre-built images from an external resource like the Container Library or Docker Hub.

Using the `pull` subcommand

When called on a native SingularityCE image like those provided by the Container Library, `pull` simply downloads the image file to your system:

```
$ singularity pull library://lolcow
```

You can also use `pull` with a `docker://` URI to reference Docker images served from a registry. In this case, `pull` does not just download an image file. Docker images are stored in layers, so `pull` must also combine those layers into a usable SingularityCE file.

```
$ singularity pull docker://sylabsio/lolcow
```

Pulling docker images may reduce reproducibility: if you were to pull a Docker image today and then wait six months and pull it again, you are not guaranteed to get the same image from docker on both occasions. If any of the source layers of the docker image has changed, the image will be altered. You can get around this by pulling docker images *by digest*, as follows:

```
$ singularity pull docker://
↪alpine@sha256:69665d02cb32192e52e07644d76bc6f25abeb5410edc1c7a81a10ba3f0efb90a
```

Note: SingularityCE will make a SIF image out of the underlying docker image; and because SIF images contain metadata (including timestamps), resulting SingularityCE images will not be bit-for-bit identical, even if they are created from docker images that were pulled by digest.

If reproducibility is a priority for you, the best option is to always build your images from the Container Library if possible.

Using the `build` subcommand

You can also use the `build` command to download pre-built images from an external resource. When using `build` you must specify a name for your container like so:

```
$ singularity build ubuntu.sif library://ubuntu

$ singularity build lolcow.sif docker://sylabsio/lolcow
```

Unlike `pull`, `build` will convert your image to the latest SingularityCE image format after downloading it. `build` is like a "Swiss Army knife" for container creation. In addition to downloading images, you can use `build` to create images from other images, or from scratch using a *definition file*. You can also use `build` to convert an image between the container formats supported by SingularityCE. To see a comparison of the SingularityCE definition file with Dockerfile, please see: *this section*.

1.3.4 Interacting with images

You can interact with images in several ways, each of which can accept image URIs in addition to local image paths.

As an example, the following command will pull a `lolcow_latest.sif` image from the Container Library:

```
$ singularity pull library://lolcow
```

Shell

The *shell* command allows you to spawn a new shell within your container and interact with it as though it were a virtual machine.

```
$ singularity shell lolcow_latest.sif
Singularity>
```

The change in prompt indicates that you have entered the container (though you should not rely on prompt forms to determine whether you are in a container or not).

Once inside of a SingularityCE container, you are the same user as you are on the host system.

```
Singularity> whoami
david

Singularity> id
uid=1000(david) gid=1000(david) groups=1000(david),4(adm),24(cdrom),27(sudo),30(dip),
↪46(plugdev),116(lpadmin),126(sambashare)
```

`shell` also works with the `library://`, `docker://`, and `shub://` URIs. This creates an ephemeral container that disappears when the shell is exited.

```
$ singularity shell library://lolcow
```

Executing Commands

The *exec* command allows you to execute a custom command within a container by specifying the image file. For instance, to execute the `cowsay` program within the `lolcow_latest.sif` container:

```
$ singularity exec lolcow_latest.sif cowsay moo
 _____
< moo >
 -----
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

`exec` also works with the `library://`, `docker://`, and `shub://` URIs. This creates an ephemeral container that executes a command and disappears.

```
$ singularity exec library://lolcow cowsay 'Fresh from the library!'
 _________________________
< Fresh from the library! >
 -------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

Running a container

SingularityCE containers contain *runscripts*. These are user-defined scripts that define the actions a container should perform when someone runs it. The runscript can be triggered with the *run* command, or simply by calling the container as though it were an executable.

```
$ singularity run lolcow_latest.sif
_______________________________
< Mon Aug 16 13:01:55 CDT 2021 >
 -------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||

$ ./lolcow_latest.sif
_______________________________
< Mon Aug 16 13:12:50 CDT 2021 >
 -------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

`run` also works with the `library://`, `docker://`, and `shub://` URIs. This creates an ephemeral container that runs and then disappears.

```
$ singularity run library://lolcow
_______________________________
< Mon Aug 16 13:12:33 CDT 2021 >
 -------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

Arguments to `run`

You can pass arguments to the runscript of a container. For example, the default runscript of the `library://alpine` container passes any arguments to a shell. We can ask the container to run `echo` command in this shell as follows:

```
$ singularity run library://alpine echo "hello"
hello
```

Because SingularityCE runscripts are evaluated shell scripts, arguments can behave slightly differently than in Docker/OCI runtimes, in the event that they contain expressions that have special meaning to the shell. Here is an illustrative example:

```
$ docker run -it --rm alpine echo "\$HOSTNAME"
$HOSTNAME

$ singularity run docker://alpine echo "\$HOSTNAME"
p700

$ singularity run docker://alpine echo "\\\$HOSTNAME"
$HOSTNAME
```

To replicate Docker/OCI behavior, you may need additional escaping or quoting of arguments.

Unlike the `run` command, the `exec` command does behave in the same manner as Docker/OCI, because it calls the specified executable directly:

```
$ singularity exec docker://alpine echo "\$HOSTNAME"
$HOSTNAME

$ singularity exec docker://alpine echo "\\\$HOSTNAME"
\$HOSTNAME
```

1.3.5 Working with Files

Files on the host are reachable from within the container:

```
$ echo "Hello from inside the container" > $HOME/hostfile.txt

$ singularity exec lolcow_latest.sif cat $HOME/hostfile.txt
Hello from inside the container
```

This example works because `hostfile.txt` exists in the user's home directory (`$HOME`). By default, SingularityCE bind mounts `$HOME`, the current working directory, and additional system locations from the host into the container.

You can specify additional directories to bind mount into your container with the `--bind` option. In the following example, the `data` directory on the host system is bind mounted to the `/mnt` directory inside the container.

```
$ echo "Drink milk (and never eat hamburgers)." > /data/cow_advice.txt

$ singularity exec --bind /data:/mnt lolcow_latest.sif cat /mnt/cow_advice.txt
Drink milk (and never eat hamburgers).
```

Pipes and redirects also work with SingularityCE commands, just like they do with normal Linux commands:

```
$ echo "Drink milk (and never eat hamburgers)." | singularity exec lolcow_latest.sif␣
↪cowsay
 _______________________________________
< Drink milk (and never eat hamburgers). >
 ---------------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

1.3.6 Building images from scratch

SingularityCE versions 3.0 and above produce immutable images in the Singularity Image File (SIF) format. This ensures reproducible and verifiable images, and allows for many extra benefits such as the ability to sign and verify your containers.

However, during testing and debugging, you may want an image format that is writable. This way you can `shell` into the image and install software and dependencies until you are satisfied that your container will fulfill your needs. For these scenarios, SingularityCE also supports the `sandbox` format (which is really just a directory).

Sandbox Directories

To build into a `sandbox` (container in a directory) use the `build --sandbox` command and option:

```
$ singularity build --sandbox ubuntu/ library://ubuntu
```

This command creates a sub-directory called `ubuntu/` with an entire Ubuntu operating system and some SingularityCE metadata in your current working directory.

You can use commands like `shell`, `exec` , and `run` with this directory just as you would with a SingularityCE image. If you pass the `--writable` option when you use your container, you can also write files within the sandbox directory (provided you have the permissions to do so).

```
$ singularity exec --writable ubuntu touch /foo

$ singularity exec ubuntu/ ls /foo
/foo
```

Converting images from one format to another

The `build` command allows you to build a new container from an existing container. This means that you can use it to convert a container from one format to another. For instance, if you have already created a sandbox (directory) and want to convert it to the Singularity Image Format, you can do so as follows:

```
$ singularity build new.sif sandbox
```

Note, however, that this approach may break reproducibility, in the event that you have altered your sandbox outside of the context of a *definition file*, so you are advised to exercise care.

SingularityCE Definition Files

For a reproducible, verifiable and production-quality container, it is recommended that you build your SIF file using a SingularityCE definition file. This also makes it easy to add files, environment variables, and install custom software, while still starting from your base of choice (e.g., the Container Library).

A definition file has a header and a body. The header determines the base container to begin with, and the body is further divided into sections that perform tasks such as software installation, environment setup, and copying files into the container from host system.

Here is an example of a definition file:

```
BootStrap: library
From: ubuntu:22.04

%post
   apt-get -y update
   apt-get -y install cowsay lolcat

%environment
   export LC_ALL=C
   export PATH=/usr/games:$PATH

%runscript
   date | cowsay | lolcat

%labels
   Author Sylabs
```

To build a container from this definition file (assuming it is a file named `lolcow.def`), you would call `build` as follows:

```
$ sudo singularity build lolcow.sif lolcow.def
```

In this example, the header tells SingularityCE to use a base Ubuntu 22.04 image from the Container Library. The other sections in this definition file are as follows:

- The `%post` section is executed within the container at build time, after the base OS has been installed. The `%post` section is therefore the place to perform installations of new libraries and applications.
- The `%environment` section defines environment variables that will be available to the container at runtime.
- The `%runscript` section defines actions for the container to take when it is executed. (These commands will therefore not be run at build time.)
- And finally, the `%labels` section allows for custom metadata to be added to the container.

This is a very small example of the things that you can do with a *definition file*. In addition to building a container from the Container Library, you can start with base images from Docker Hub and use images directly from official repositories such as Ubuntu, Debian, CentOS, Arch, and BusyBox. You can also use an existing container on your host system as a base. Definition files also support *"templating"*: the ability to pass values from the command-line, or from a definitions file, that will replace placeholders in the definition file at build time.

If you want to build SingularityCE images but you don't have administrative (root) access on your build system, you can build images using the Remote Builder.

This quickstart document just scratches the surface of all of the things you can do with SingularityCE!

If you need additional help or support, contact the Sylabs team: https://www.sylabs.io/contact/

1.3.7 SingularityCE on a shared resource

Perhaps you are a user who wants a few talking points and background to share with your administrator. Or maybe you are an administrator who needs to decide whether to install SingularityCE.

This document and the accompanying administrator documentation provide answers to many common questions.

If you need to request an installation from your administrator, you may decide to draft a message similar to this:

```
Dear shared resource administrator,

We are interested in having SingularityCE (https://www.sylabs.io/docs/)
installed on our shared resource. SingularityCE containers will allow us to
build encapsulated environments, meaning that our work is reproducible and
we are empowered to choose all dependencies including libraries, operating
system, and custom software. SingularityCE is already in use on many of the
top HPC centers around the world. Examples include:

    Texas Advanced Computing Center
    GSI Helmholtz Center for Heavy Ion Research
    Oak Ridge Leadership Computing Facility
    Purdue University
    National Institutes of Health HPC
    UFIT Research Computing at the University of Florida
    San Diego Supercomputing Center
    Lawrence Berkeley National Laboratory
    University of Chicago
    McGill HPC Centre/Calcul Québec
    Barcelona Supercomputing Center
    Sandia National Lab
    Argonne National Lab

Importantly, it has a vibrant team of developers, scientists, and HPC
administrators that invest heavily in the security and development of the
software, and are quick to respond to the needs of the community. To help
learn more about SingularityCE, I thought these items might be of interest:

    - Security: A discussion of security concerns is discussed at
    https://www.sylabs.io/guides/4.0/admin-guide/admin_quickstart.html

    - Installation:
    https://www.sylabs.io/guides/4.0/admin-guide/installation.html

If you have questions about any of the above, you can contact the open
source list (https://groups.google.com/g/singularity-ce), join the open
source slack channel (singularityce.slack.com), or contact the organization
that supports SingularityCE directly (sylabs.io/contact). I can do my best
to facilitate this interaction if help is needed.

Thank you kindly for considering this request!

Best,

User
```

1.4 Security in SingularityCE

1.4.1 Security Policy

If you suspect you have found a vulnerability in SingularityCE, we want to work with you so that it can be investigated, fixed, and disclosed in a responsible manner. Please follow the steps in our published Security Policy, which begins with contacting us privately via security@sylabs.io.

Sylabs discloses vulnerabilities found in SingularityCE through public CVE reports, as well as notifications on our community channels. We encourage all users to monitor new releases of SingularityCE for security information. Security patches are applied to the latest open-source release.

SingularityPRO is a professionally curated and licensed version of SingularityCE that provides added security, stability, and support beyond that offered by the open-source project. Security and bug-fix patches are backported to select versions of SingularityPRO, so that they can be deployed long-term where required. PRO users receive security fixes as detailed in the Sylabs Security Policy.

1.4.2 Background

SingularityCE grew out of the need to implement a container platform that was suitable for use on shared systems, such as HPC clusters. In these environments, multiple people typically need to access the same shared resource. User accounts, groups, and standard file permissions limit their access to data and devices, and prevent them from disrupting or accessing others' work.

To provide security in these environments a container needs to run as the user who starts it on the system. Before the widespread adoption of the Linux user namespace, only a privileged user could perform the operations which are needed to run a container. A default Docker installation uses a root-owned daemon to start containers, and users can request that the daemon start a container on their behalf. However, coordinating a daemon with other schedulers is difficult and, since the daemon is privileged, users can ask it to carry out actions that they wouldn't normally have permission to carry out themselves.

When a user runs a container with SingularityCE, it is started as a normal process running under the user's account. Standard file permissions and other security controls based on user accounts, groups, and processes apply. In a default installation, SingularityCE uses a setuid starter binary to perform only the specific tasks needed to setup the container.

1.4.3 Setuid & User Namespaces

Using a setuid binary to run container setup operations is essential to support containers on older Linux distributions, such as CentOS 6, that were previously common in HPC and enterprise. Newer distributions have support for 'unprivileged user namespace creation'. This means a normal user can create a user namespace, in which most setup operations needed to run a container can be run, unprivileged.

Security Implications of Unprivileged User Namespaces

Warning: If you rely on the ECL or other container execution limits, you must disable unprivileged user namespace creation on your systems.

When unprivileged user namespace creation is allowed on a system, a user can supply and use their own unprivileged installation of Singularity or another container runtime. They may also be able to use standard system tools such as `unshare`, `nsenter`, and FUSE mounts to access / execute arbitrary containers without installing any runtime. Both

of these approaches will allow users to bypass any restrictions that have been set in a system-wide installation of SingularityCE. These include:

- The `allow container` and `limit container` directives in `singularity.conf`.
- The Execution Control List, which restricts execution of SIF container images via signature checks.

Note also that SingularityCE's *–oci* mode is an unprivileged runtime that requires unprivileged user namespace creation. It does not implement the container restrictions that cannot be effectively enforced when unprivileged user namespaces are available.

If your primary security concern is that of restricting the containers which users can execute, you should use singularity in setuid mode, and ensure unprivileged user namespace creation is disabled on the host.

Configuration and Limitations of User Namespace Mode

SingularityCE supports running containers without setuid, using user namespaces. It can be compiled with the `--without-setuid` option, or `allow setuid = no` can be set in `singularity.conf` to enable this. In this mode *all* operations run as the user who starts the `singularity` program. However, there are some disadvantages to this approach:

- SIF and other single-file container images cannot be mounted directly. The container image must be extracted to a directory on disk to run. This impacts the speed of execution. As a result, workloads accessing large numbers of small files (as is the case with python application startup, for example) do not benefit from the reduced metadata load on the filesystem an image file provides.

  SingularityCE 3.10 introduces experimental functionality to avoid this image-extraction by mounting the SIF container using `squashfuse`, if available on your system. You can enable this with the `--sif-fuse` flag, or by setting `sif fuse` in `singularity.conf`.
- Replacing direct kernel mounts with a FUSE approach is likely to cause a significant reduction in performance.
- The effectiveness of signing and verifying container images is reduced as, when extracted to a directory, modification of files is possible and verification of the image's original signature cannot be performed.
- Encryption is not supported. SingularityCE leverages kernel LUKS2 mounts to run encrypted containers without writing a decrypted version of their content to disk.
- Some hold the opinion that vulnerabilities in kernel user namespace code could have greater impact than vulnerabilities confined to a single piece of setuid software, and are therefore reluctant to enable unprivileged user namespace creation.
- Limitations on container execution by location, valid signatures, user/group cannot be enforced.

Because of the points above, the default mode of operation of SingularityCE uses a setuid binary. Sylabs aims to reduce the circumstances that require this as new functionality is developed and reaches commonly deployed Linux distributions.

1.4.4 Runtime & User Privilege Model

While other runtimes have aimed to tackle security concerns by sandboxing containers executing as the `root` user so that they cannot affect the host system, SingularityCE has adopted a different security model:

- Containers should be run as an unprivileged user.
- The user should never be able to elevate their privileges inside the container to gain control over the host.
- All permission restrictions on the user outside of a container should apply inside the container, as well.

- Favor integration over isolation: a user is allowed to easily use host resources such as GPUs, network filesystems, and high speed interconnects. The process ID space, network, etc., are not isolated in separate namespaces by default.

To accomplish this, SingularityCE uses a number of Linux kernel features. The container file system is mounted using the `nosuid` option, and processes are started with the `PR_NO_NEW_PRIVS` flag set. This means that even if you run `sudo` inside your container, you won't be able to change to another user, or gain root privileges by other means.

If you do require the additional isolation of the network, devices, PIDs, etc., which other runtimes provide, SingularityCE can make use of additional namespaces and functionality such as seccomp and cgroups.

1.4.5 Singularity Image Format (SIF)

SingularityCE uses SIF as its default container format. A SIF container is a single file, which makes it easy to manage and distribute. Inside the SIF file, the container filesystem is held in a SquashFS object. By default, we mount the container filesystem directly using SquashFS. On a network filesystem, this means that reads from the container are data-only. Metadata operations happen locally, speeding up workloads that involve many small files.

Holding the container image in a single file also enables unique security features. The container filesystem is immutable, and can be signed. The signature travels as part of the SIF image itself so that it is always possible to verify that the image has not been tampered with or corrupted.

We use private PGP keys to create a container signature, and the corresponding public keys to verify the container. Verification of signed containers happens automatically in `singularity pull` commands against the Singularity Container Services (SCS) Library. The SCS keystore makes it easier to share and obtain public keys for container verification.

A container may be signed once, by a trusted individual who approves its use. It could also be signed with multiple keys to signify it has passed each step in a CI/CD QA & Security process. In setuid mode, SingularityCE can be configured with an execution control list (ECL). The ECL requires the presence of one or more valid signatures, to limit execution to approved containers on systems that have unprivileged user namespace creation disabled.

In SingularityCE 3.4 and above, the root filesystem of a container (stored in the SquashFS partition of SIF) can be encrypted. As a result, everything inside the container becomes inaccessible without the correct key or passphrase. The content of the container then remains private, even if the SIF file is shared in public.

Encryption and decryption are performed using the Linux kernel's LUKS2 feature. This is the same technology routinely used for full disk encryption. The encrypted container is mounted directly through the kernel. Unlike other container formats, the encrypted container is run without ever decrypting its contents to disk.

1.4.6 Configuration & Runtime Options

System administrators who manage SingularityCE can use configuration files to set security restrictions, grant or revoke a user's capabilities, manage resources, authorize containers, etc.

For example, the ecl.toml configuration file allows blacklisting and whitelisting of containers.

Documentation for administrators about configuration files and their parameters is available here.

When running a container as root, SingularityCE can apply hardening rules using cgroups, seccomp, and apparmor. See *here* for details on these options.

CHAPTER

TWO

BUILDING CONTAINERS

Learn how to write a definition file that can be used to build a container. Understand the environment within a build, how to perform remote builds, and how to use the `--fakeroot` feature to build as a non-root user.

2.1 Build a Container

The `build` command is the "Swiss army knife" of container creation. You can use it to download and assemble existing containers from external resources like the Container Library and Docker Hub. You can use it to convert containers between the formats supported by SingularityCE. And you can use it in conjunction with a *SingularityCE definition* file to create a container from scratch and customized it to fit your needs.

2.1.1 Overview

The `build` command accepts a target as input and produces a container as output.

The type of target given determines the method that `build` will use to create the container. It can be one of the following:

- URI beginning with **library://** to build from the Container Library
- URI beginning with **docker://** to build from Docker Hub
- URI beginning with **shub://** to build from Singularity Hub
- path to an **existing container** on your local machine
- path to a **directory** to build from a sandbox
- path to a *SingularityCE definition file*

`build` can produce containers in two different formats, which can be specified as follows:

- a compressed read-only **Singularity Image File (SIF)** format, suitable for production *(default)*
- a writable **(ch)root directory** called a sandbox, for interactive development (`--sandbox` option)

Because `build` can accept an existing container as a target and create a container in either supported format, you can use it to convert existing containers from one format to another.

2.1.2 Downloading an existing container from the Container Library

You can use the `build` command to download a container from the Container Library:

```
$ sudo singularity build lolcow.sif library://lolcow
```

The first argument (`lolcow.sif`) specifies the path and name for your container. The second argument (`library://lolcow`) gives the Container Library URI from which to download. By default, the container will be converted to a compressed, read-only SIF. If you want your container in a writable format, use the `--sandbox` option.

2.1.3 Downloading an existing container from Docker Hub

You can use `build` to download layers from Docker Hub and assemble them into SingularityCE containers.

```
$ sudo singularity build lolcow.sif docker://sylabsio/lolcow
```

2.1.4 Creating writable `--sandbox` directories

If you want to create a container within a writable directory (called a *sandbox*) you can do so with the `--sandbox` option. It's possible to create a sandbox without root privileges, but to ensure proper file permissions, it is recommended to do so as root:

```
$ sudo singularity build --sandbox lolcow/ library://lolcow
```

The resulting directory operates just like a container in a SIF file. To make persistent changes within the sandbox container, use the `--writable` flag when you invoke your container. It's a good idea to do this as root to ensure you have permission to access the files and directories that you want to change.

```
$ sudo singularity shell --writable lolcow/
```

2.1.5 Converting containers from one format to another

If you already have a container saved locally, you can use it as a target to build a new container. This allows you convert containers from one format to another. For example, if you had a sandbox container called `development/` and you wanted to convert it to SIF container called `production.sif`, you could do so as follows:

```
$ sudo singularity build production.sif development/
```

Use care when converting a sandbox directory to the default SIF format. If changes were made to the writable container before conversion, there is no record of those changes in the SingularityCE definition file, which compromises the reproducibility of your container. It is therefore preferable to build production containers directly from a SingularityCE definition file, instead.

2.1.6 Building containers from SingularityCE definition files

SingularityCE definition files are the most powerful type of target when building a container. For detailed information on writing SingularityCE definition files, please see the *Container Definitions documentation*. Suppose you already have the following container definition file, called `lolcow.def`, and you want to use it to build a SIF container:

```
Bootstrap: docker
From: ubuntu:22.04

%post
    apt-get -y update
    apt-get -y install cowsay lolcat

%environment
    export LC_ALL=C
    export PATH=/usr/games:$PATH

%runscript
    date | cowsay | lolcat
```

You can do so with the following command.

```
$ sudo singularity build lolcow.sif lolcow.def
```

In this case, we're running `singularity build` with `sudo` because installing software with `apt-get`, as in the `%post` section, requires the root privileges. By default, when you run SingularityCE, you are the same user inside the container as on the host machine. Using `sudo` on the host, to acquire root privileges, ensures we can use `apt-get` as root inside the container.

If you aren't able or do not wish to use `sudo` when building a container, SingularityCE offers several other options: `--remote` builds, a `--fakeroot` mode, and limited unprivileged builds using `proot`.

`--remote` builds

Singularity Container Services and Singularity Enterprise provide a *Remote Build Service*. This service can perform a container build, as the root user, inside a secure single-use virtual machine.

Remote builds do not have the system requirements of `--fakeroot` builds, or the limitations of unprivileged `proot` builds. They are a convenient way to build SingularityCE containers on systems where `sudo` rights are not available.

To perform a remote build, you will need a Singularity Container Services account. (If you do not already have an account, you can create one on the site.) Once you have an account, ensure you are logged in from your command-line environment by running:

```
$ singularity remote login
```

You can then add the `--remote` flag to your build command:

```
$ singularity build --remote lolcow.sif lolcow.def
```

The build will be sent to the remote build service, and the progress and output of your build will be displayed on your local machine. When the build is complete, the resulting SIF container image will be downloaded to your machine.

`--fakeroot` builds

A build run with the `--fakeroot` flag uses certain Linux kernel features to enable you to run as an emulated, 'fake' root user inside the container, while running as your regular user (and not as root) on the host system.

The `--fakeroot` feature has particular requirements in terms of the capabilities and configuration of the host system. This is covered further in the *fakeroot* fakeroot section of this user guide, as well as in the admin guide.

If your system is configured for `--fakeroot` support, then you can run the above build without using `sudo`, by adding the `--fakeroot` flag:

```
$ singularity build --fakeroot lolcow.sif lolcow.def
```

Unprivilged `proot` builds

SingularityCE 3.11 introduces the ability to run some definition file builds without `--fakeroot` or `sudo`. This is useful on systems where you cannot `sudo`, and the administrator cannot perform the configurations necessary for `--fakeroot` support.

Unprivileged `proot` builds are automatically performed when proot is available on the system `PATH`, and `singularity build` is run by a non-root user against a definition file:

```
$ singularity build lolcow.sif lolcow.def
INFO:    Using proot to build unprivileged. Not all builds are supported. If build fails,
↪ use --remote or --fakeroot.
INFO:    Starting build...
```

Unprivileged builds that use `proot` have limitations, because `proot`'s emulation of the root user is not complete. In particular, such builds:

- Do not support `arch` / `debootstrap` / `yum` / `zypper` bootstraps. Use `localimage`, `library`, `oras`, or one of the `docker`/`oci` sources.
- Do not support `%pre` and `%setup` sections of definition files.
- Run the `%post` sections of a build in the container as an emulated root user.
- Run the `%test` section of a build as the non-root user, like `singularity test`.
- Are subject to any restrictions imposed in `singularity.conf`.
- Incur a performance penalty due to the``ptrace``-based interception of syscalls used by `proot`.
- May fail if the `%post` script requires privileged operations that `proot` cannot emulate.

Generally, if your definition file starts from an existing SIF/OCI container image, and adds software using system package managers, an unprivileged proot build is appropriate. If your definition file compiles and installs large complex software from source, you may wish to investigate `--remote` or `--fakeroot` builds instead.

2.1.7 Building encrypted containers

Starting with SingularityCE 3.4.0, it is possible to build and run encrypted containers. The containers are decrypted at runtime entirely in kernel space, meaning that no intermediate decrypted data is ever written to disk. See *encrypted containers* for more details.

2.1.8 Build options

`--builder`

SingularityCE 3.0 introduces the option to perform a remote build. The `--builder` option allows you to specify a URL to a different build service. For instance, you may need to specify a URL pointing to an on-premises installation of the remote builder. This option must be used in conjunction with `--remote`.

`--detached`

When used in combination with the `--remote` option, the `--detached` option will detach the build from your terminal and allow it to build in the background without echoing any output to your terminal.

`--encrypt`

Specifies that SingularityCE should use a secret saved in either the `SINGULARITY_ENCRYPTION_PASSPHRASE` or `SINGULARITY_ENCRYPTION_PEM_PATH` environment variable to build an encrypted container. See *encrypted containers* for more details.

`--fakeroot`

Gives users a way to build containers without root privileges. See *the fakeroot feature* for details.

`--force`

The `--force` option will delete and overwrite an existing SingularityCE image without presenting the normal interactive confirmation prompt.

`--json`

The `--json` option will force SingularityCE to interpret a given definition file as JSON.

`--library`

This command allows you to set a different image library. (The default library is "https://library.sylabs.io")

--notest

If you don't want to run the `%test` section during the container build, you can skip it using the `--notest` option. For instance, you might be building a container intended to run in a production environment with GPUs, while your local build resource does not have GPUs. You want to include a `%test` section that runs a short validation, but you don't want your build to exit with an error because it cannot find a GPU on your system. In such a scenario, passing the `--notest` flag would be appropriate.

--passphrase

This flag allows you to pass a plaintext passphrase to encrypt the container filesystem at build time. See *encrypted containers* for more details.

--pem-path

This flag allows you to pass the location of a public key to encrypt the container file system at build time. See *encrypted containers* for more details.

--remote

SingularityCE 3.0 introduces the ability to build a container on an external resource running a remote builder. (The default remote builder is located at "https://cloud.sylabs.io/builder".)

--sandbox

Build a sandbox (container in a directory) instead of the default SIF format.

--section

Instead of running the entire definition file, only run a specific section or sections. This option accepts a comma-delimited string of definition file sections. Acceptable arguments include `all`, `none` or any combination of the following: `setup`, `post`, `files`, `environment`, `test`, `labels`.

Under normal build conditions, the SingularityCE definition file is saved into a container's metadata so that there is a record of how the container was built. The `--section` option may render this metadata inaccurate, compromising reproducibility, and should therefore be used with care.

--update

You can build into the same sandbox container multiple times (though the results may be unpredictable, and under most circumstances, it would be preferable to delete your container and start from scratch).

By default, if you build into an existing sandbox container, the `build` command will prompt you to decide whether or not to overwrite existing container data. Instead of this behavior, you can use the `--update` option to build *into* an existing container. This will cause SingularityCE to skip the definition-file's header, and build any sections that are in the definition file into the existing container.

The `--update` option is only valid when used with sandbox containers.

`--nv`

This flag allows you to mount the NVIDIA CUDA libraries from your host environment into your build environment. Libraries are mounted during the execution of `post` and `test` sections.

`--rocm`

This flag allows you to mount the AMD Rocm libraries from your host environment into your build environment. Libraries are mounted during the execution of `post` and `test` sections.

`--bind`

This flag allows you to mount a directory, file or image during build. It works the same way as `--bind` for the `shell`, `exec` and `run` subcommands of SingularityCE, and can be specified multiple times. See *user defined bind paths*. Bind mounts occur during the execution of `post` and `test` sections.

`--writable-tmpfs`

This flag will run the `%test` section of the build with a writable `tmpfs` overlay filesystem in place. This allows the tests to create files, which will be discarded at the end of the build. Other portions of the build do not use this temporary filesystem.

2.1.9 More Build topics

- If you want to **customize the cache location** (where Docker layers are downloaded on your system), specify Docker credentials, or apply other custom tweaks to your build environment, see *build environment*.
- If you want to make internally **modular containers**, check out the Getting Started guide here.
- If you want to **build your containers** on the Remote Builder, (because you don't have root access on a Linux machine, or you want to host your container on the cloud), check out this site.
- If you want to **build a container with an encrypted file system** consult the SingularityCE documentation on encryption *here*.

2.2 Definition Files

A SingularityCE Definition File (or "def file" for short) is like a set of blueprints explaining how to build a custom container. It includes specifics about the base OS to build or the base container to start from, as well as software to install, environment variables to set at runtime, files to add from the host system, and container metadata.

2.2.1 Overview

A SingularityCE Definition file is divided into two parts:

1. **Header**: The Header describes the core operating system to build within the container. Here you will configure the base operating system features needed within the container. You can specify the Linux distribution, the specific version, and the packages that must be part of the core install (borrowed from the host system).
2. **Sections**: The rest of the definition is comprised of sections, (sometimes called scriptlets or blobs of data). Each section is defined by a `%` character followed by the name of the particular section. All sections are optional, and a def file may contain more than one instance of a given section. Sections that are executed at build time are executed with the `/bin/sh` interpreter and can accept `/bin/sh` options. Similarly, sections that produce scripts to be executed at runtime can accept options intended for `/bin/sh`.

For more in-depth and practical examples of def files, see the Sylabs examples repository.

For a direct comparison between Dockerfiles and SingularityCE definition files, you can jump directly to *the relevant section* in the documentation.

2.2.2 Header

The header should be located at the beginning of the def file. It tells SingularityCE about the base operating system that it should use to build the container. It is composed of several keywords.

The only keyword that is required for every type of build is `Bootstrap`. It determines the *bootstrap agent* that will be used to create the base operating system you want to use. For example, the `library` bootstrap agent will pull a container from the SingularityCE Container Library as a base. Similarly, the `docker` bootstrap agent will pull docker layers from Docker Hub as a base OS from which to start your image.

Starting with SingularityCE 3.2, the `Bootstrap` keyword needs to be the first entry in the header section. This breaks compatibility with older versions, which allowed the parameters of the header to appear in any order.

Depending on the value assigned to `Bootstrap`, other keywords may also be valid in the header. For example, when using the `library` bootstrap agent, the `From` keyword becomes valid. Here is an example that uses the `From` keyword to build a Debian container from the Container Library:

```
Bootstrap: library
From: debian:7
```

A def file that uses an official mirror to install CentOS 7 might look like this:

```
Bootstrap: yum
OSVersion: 7
MirrorURL: http://mirror.centos.org/centos-%{OSVERSION}/%{OSVERSION}/os/$basearch/
Include: yum
```

Each bootstrap agent enables its own options and keywords, which you can read about, and see examples of, in the *appendix*.

Preferred bootstrap agents

- *library* (images hosted on the SingularityCE Container Library)
- *docker* (images hosted on Docker Hub)
- *shub* (images hosted on Singularity Hub)
- *oras* (images from supported OCI registries)
- *scratch* (a flexible option for building a container from scratch)

Other bootstrap agents

- *localimage* (images saved on your machine)
- *yum* (yum-based systems such as CentOS and Scientific Linux)
- *debootstrap* (apt-based systems such as Debian and Ubuntu)
- oci (bundle compliant with OCI Image Specification)
- oci-archive (tar files obeying the OCI Image Layout Specification)
- docker-daemon (images managed by the locally running docker daemon)
- docker-archive (saved docker images)
- *arch* (Arch Linux)
- *busybox* (BusyBox)
- *zypper* (zypper-based systems such as SUSE and openSUSE)

SIF Image Verification / Fingerprints Header

If the bootstrap image is in the SIF format, then verification will be performed at build time. This verification checks whether the image has been signed. If it has been signed, the integrity of the image is checked, and the signatures matched against public keys if available. This process is equivalent to running `singularity verify` on the bootstrap image.

By default, a failed verification (e.g. against an unsigned image, or one that has been modified after signing) will produce a warning, but the build will continue.

To make it a requirement that the bootstrap image verifies correctly and has been signed by one or more keys, you can use the `Fingerprints:` header introduced in SingularityCE 3.7.

```
Bootstrap: localimage
From: test.sif
Fingerprints: 12045C8C0B1004D058DE4BEDA20C27EE7FF7BA84,
↪22045C8C0B1004D058DE4BEDA20C27EE7FF7BA84
```

If, at build time, the image is not signed with keys corresponding to *all* of the listed fingerprints, the build will fail.

The `Fingerprints:` header can be used with bootstrap agents that provide a SIF image. The `library` agent always retrieves a SIF image. The `localimage` agent *can* be used to refer to SIF images, which will work correctly with the `Fingerprints:` header, but also to other types of images, which will not.

The `Fingerprints:` header has no effect if the bootstrap image is not in SIF format.

Note: The verification occurs before the bootstrap image is extracted into a temporary directory for the build process. The fingerprint check ensures the correct image was retrieved for the build, but does not protect against malicious changes that could be made during the build process on an already-compromised machine.

2.2.3 Sections

The main content of the bootstrap file is broken into sections. Different sections add different content, or execute commands at different times during the build process. Note that if any command fails, the build process will halt.

Here is an example definition file that uses every available section. We will discuss each section in turn, below. It is not necessary to include every section (or any sections at all) within a def file. Furthermore, multiple sections of the same name can be included and will be appended to one another during the build process.

```
Bootstrap: library
From: ubuntu:22.04
Stage: build

%setup
    touch /file1
    touch ${SINGULARITY_ROOTFS}/file2

%files
    /file1
    /file1 /opt

%environment
    export LISTEN_PORT=54321
    export LC_ALL=C

%post
    apt-get update && apt-get install -y netcat
    NOW=`date`
    echo "export NOW=\"${NOW}\"" >> $SINGULARITY_ENVIRONMENT

%runscript
    echo "Container was created $NOW"
    echo "Arguments received: $*"
    exec echo "$@"

%startscript
    nc -lp $LISTEN_PORT

%test
    grep -q NAME=\"Ubuntu\" /etc/os-release
    if [ $? -eq 0 ]; then
        echo "Container base is Ubuntu as expected."
    else
        echo "Container base is not Ubuntu."
        exit 1
    fi
```

(continues on next page)

(continued from previous page)

```
%labels
    Author myuser@example.com
    Version v0.0.1

%help
    This is a demo container used to illustrate a def file that uses all
    supported sections.
```

Although the order of the sections in the def file is unimportant, they have been documented below in the order of their execution during the build process for ease of understanding.

%setup

During the build process, commands in the `%setup` section are first executed on the host system *outside* of the container, after the base OS has been installed. You can reference the container file system with the `$SINGULARITY_ROOTFS` environment variable in the `%setup` section.

Note: Be careful with the `%setup` section! This scriptlet is executed outside of the container on the host system itself, and is executed with elevated privileges when you run the build as `root` or with `sudo`. Commands in `%setup` can alter and potentially damage the host.

Moreover, whether the code in `%setup` runs successfully and correctly will depend on the configuration of the host system. That is exactly the kind of environment-dependency that containerization is meant to circumvent, in the first place.

You should avoid `%setup` wherever possible, and work inside the container in the `%post` block instead.

Consider the example from the definition file above:

```
%setup
    touch /file1
    touch ${SINGULARITY_ROOTFS}/file2
```

Here, `file1` is created at the root of the file system **on the host**. We'll use `file1` to demonstrate the usage of the `%files` section below. `file2`, on the other hand, is created at the root of the file system **within the container**.

More recent versions of SingularityCE provide the `%files` section, which is a safer alternative to copying files from the host system into the container during the build process.

%files

The `%files` section allows you to copy files into the container with greater safety than using the `%setup` section. Its general form is:

```
%files [from <stage>]
    <source> [<destination>]
    ...
```

Each line is a `<source>` and `<destination>` pair. The `<source>` is either:

1. A valid path to a file or directory on your host system
2. A valid *file.Match* pattern matching one or more files or directories on your host system

3. A valid path in a previous stage of the build

The <destination> is a path inside the current container. If the <destination> path is omitted it will be assumed to be the same as <source>. To show how copying from the host system works, let's consider the example from the definition file above:

```
%files
    /file1
    /file1 /opt
```

This %files scriptlet will copy file1 from the root of the host filesystem to the root of the container filesystem, and then make a second copy of file1 inside /opt within the container filesystem (i.e., at /opt/file1) .

Copying Multiple Files with Patterns

To copy multiple files or directories at a time, you can specify a pattern as the <source> for a %files line. SingularityCE matches patterns to filenames using the Go file.Match syntax:

```
'*'         matches any sequence of non-Separator characters
'?'         matches any single non-Separator character
'[' [ '^' ] { character-range } ']'
            character class (must be non-empty)
c           matches character c (c != '*', '?', '\\', '[')
'\\' c      matches character c

character-range:
c           matches character c (c != '\\', '-', ']')
'\\' c      matches character c
lo '-' hi   matches character c for lo <= c <= hi
```

(See: https://pkg.go.dev/path/filepath#Match)

Some simple examples:

- myfile* will match all files that have a name beginning with myfile. E.g. myfile-blue and myfileRed.
- experiment??? will match all files that have a name beginning with experiment and followed by any three characters. It will match experiment001,``experiment002``, and experimentABC, but not experimentA.
- document[0-9] will match document1, but not documentA or document01.

Copying Files Between stages

Files can also be copied from other stages in a multi stage build, by providing the source location in the previous stage and the destination in the current container.

```
%files from stage_name
  /root/hello /bin/hello
```

The only difference in behavior between copying files from your host system and copying them from previous build stages is that in the former case, symbolic links are *followed*, while in the latter case, symbolic links are *preserved as symbolic links*.

Files in the %files section are always copied before the %post section is executed, so that they are available during the build and configuration process.

%app*

In some circumstances, it may be redundant to build different containers for each app with nearly equivalent dependencies. SingularityCE supports installing apps within internal modules based on the concept of the Scientific Filesystem (SCIF). More information on defining and using SCIF Apps can be found *here*.

%post

This section is where you can download files from the internet with tools like `git` and `wget`, install new software and libraries, write configuration files, create new directories, etc.

The commands in the `%post` section run in a clean environment. Environment variables from the host are not passed into the build. To pass values into a build you should use the *templating / build-args support* introduced in SingularityCE 4.0.

Consider the `%post` section from the example definition file above:

```
%post
    apt-get update && apt-get install -y netcat

    NOW=`date`
    echo "export NOW=\"${NOW}\"" >> $SINGULARITY_ENVIRONMENT
```

This `%post` scriptlet uses the Ubuntu package manager `apt` to update the container and install the program `netcat` (that will be used in the `%startscript` section below).

The scriptlet also sets an environment variable called `NOW` to the current date and time. It then writes an `export` statement for `NOW` into the file that is pointed to by `$SINGULARITY_ENVIRONMENT`. This demonstrates how to set container environment variables (available when the container is run), that are not known when the definition file is written. Because the value of `$NOW` is only known when the `date` command in the `%post` scriptlet is run, it cannot be added to the `%environment` section.

Directing output into `$SINGULARITY_ENVIRONMENT` will cause it to be written to a shell script file called `/.singularity.d/env/91-environment.sh` that will be sourced by the container at runtime. The `export NAME=VALUE` syntax is needed to make the environment variable available to all processes that are started in the container.

Note: Variables set in the `%post` section through `$SINGULARITY_ENVIRONMENT` take precedence over those added via `%environment`.

The `%post` scriptlet will run under `sh` or `bash` by default. You can change the shell or interpreter that the scriptlet runs under by using a `-c <shell>` argument on the `%post` line, e.g:

```
%post -c /bin/zsh
   ...
```

In the `%post` section above, the scriptlet will be run by the zsh shell installed at `/bin/zsh` in the container. The requested shell must be present in the base image that was bootstrapped.

Note: Unlike the `%test` and `%runscript` sections, the `%post` section does not support hashbang lines (`#!`) for specifying a custom shell. The `-c <shell>` argument must be used instead.

%test

The `%test` section runs at the very end of the build process, and can be used to validate the container using methods of your choosing. You can also execute this scriptlet through the container itself, using the `test` command.

Consider the `%test` section from the example definition file above:

```
%test
    grep -q NAME=\"Ubuntu\" /etc/os-release
    if [ $? -eq 0 ]; then
        echo "Container base is Ubuntu as expected."
    else
        echo "Container base is not Ubuntu."
        exit 1
    fi
```

This (somewhat trivial) script tests whether the base OS is Ubuntu. You can use the `%test` section to test whether binaries were appropriately downloaded and built, or whether software works as expected on custom hardware. If you want to build a container without running the `%test` section (for example, if your build system does not have the same hardware that will be used in your production environment), you can do so by passing the `--notest` flag to the build command:

```
$ sudo singularity build --notest my_container.sif my_container.def
```

Running the test command on a container built with this def file yields the following:

```
$ singularity test my_container.sif
Container base is Ubuntu as expected.
```

One common use of the `%test` section is to run a quick check that the programs you installed in the container are indeed present.

Suppose you've installed the program `samtools`, by adding it to the list of packages passed to `apt-get install` in the `%post` section:

```
%post
    apt-get update && apt-get install -y netcat samtools
    NOW=`date` echo "export NOW=\"${NOW}\"" >>
    $SINGULARITY_ENVIRONMENT
```

`samtools` prints a usage message when run without any options, so you might decide to test that it can be run by writing the following in the `%test` section:

```
%test
    echo 'Looking for samtools...'
    samtools
```

If `samtools` is not successfully installed in the container, then `singularity build` (if run without the `--notest` flag) will produce an error (such as `samtools: not found`) during the test phase of the build, and running `singularity test` will produce the same error.

The problem with this approach is that, like many other programs, `samtools` returns a non-zero error code when run without its mandatory options. So, while the `%test` section we just wrote will print the usage message of `samtools` if `samtools` has been installed, it will also report the error code (reflecting the absence of mandatory options to `samtools`), which is probably not what we want in this case.

A better approach would therefore be to run `samtools` with the `version` option, and check that the output is what we expected. Here, we do this by running `grep` on the output and checking that the version number begins with "1":

```
%test
    echo 'Looking for samtools...'
    ( samtools --version | grep -q 'samtools 1' ) && echo 'Success!'
```

Because the `%test` section is a shell scriptlet, complex tests are possible. Remember that your scriptlet should be written so it exits with a non-zero error code if the test encounters a problem.

The `%test` scriptlet will run under `sh` or `bash` by default. You can change the shell or interpreter that the test runs under by using a custom hashbang (`#!`) as the first line in your `%test` section:

```
%test
   #!/bin/zsh

   echo "$(readlink /proc/$$/exe) is our shell"
```

In the `%test` section above, the `#!/bin/zsh` means that the test code will be run by the zsh shell installed at `/bin/zsh`. The `echo` statement given above will display the shell that is running the script, confirming that this works.

A custom hashbang runs the specified shell from the container filesystem, not the host. Therefore, `zsh` must be installed in the *container*, and since `zsh` is not built into the base Ubuntu image, it would have to be installed as part of the `%post` section for this `%test` code to work properly.

%environment

The `%environment` section allows you to define environment variables that will be set at runtime. Note that these variables are made available in the container at runtime, but not at build time. This means that if you need the same variables during the build process, you should also define them in your `%post` section. Specifically:

- **during build**: The `%environment` section is written to a dedicated file in the container metadata directory. This file is not sourced.
- **during runtime**: The file in the container metadata directory is sourced.

You should use the same conventions that you would use in a `.bashrc` or `.profile` file. Consider the `%environment` section from the example definition file above:

```
%environment
    export LISTEN_PORT=54321
    export LC_ALL=C
```

The `$LISTEN_PORT` variable will be used in the `%startscript` section of the same example, discussed below. The `$LC_ALL` variable is useful for many programs (especially those written in Perl) that expect a locale to be set.

After building this container, you can use a command like the following one to verify that the environment variables have been set appropriately at runtime:

```
$ singularity exec my_container.sif env | grep -E 'LISTEN_PORT|LC_ALL'
LISTEN_PORT=54321
LC_ALL=C
```

To set a default value for a variable in the `%environment` section, but adopt the value of a host environment variable if it is set, use the following syntax:

```
%environment
   FOO=${FOO:-'default'}
```

The value of FOO in the container will take the value of FOO on the host, or default if FOO is not set on the host or if --cleanenv / --containall have been specified.

Note: Variables added to the $SINGULARITY_ENVIRONMENT file in the %post section will take precedence over variables set in the %environment section.

See *Environment and Metadata* for more information about the SingularityCE container environment.

%startscript

Similar to the %runscript section, the contents of the %startscript section are written to a dedicated file within the container at build time. This file is executed when the instance start command is issued.

Consider the %startscript section from the example definition file above:

```
%startscript
    nc -lp $LISTEN_PORT
```

Here, the netcat (nc) program is used to listen for TCP traffic on the port indicated by the $LISTEN_PORT variable (set in the %environment section, above). The script can be invoked as follows:

```
$ singularity instance start my_container.sif instance1
INFO:    instance started successfully

$ netstat -ln | grep 54321
tcp        0      0 0.0.0.0:54321           0.0.0.0:*               LISTEN

$ singularity instance stop instance1
Stopping instance1 instance of /home/vagrant/my_container.sif (PID=19035)
```

%runscript

The contents of the %runscript section are written to a dedicated file within the container that is executed when the container image is run (either via the singularity run command or by *executing the container directly* as a command). When the container is invoked, arguments following the container name are passed to the runscript. This means that you can (and should) process arguments within your runscript.

Consider the %runscript section from the example definition file above:

```
%runscript
    echo "Container was created $NOW"
    echo "Arguments received: $*"
    exec echo "$@"
```

In this runscript, the time that the container was created is echoed via the $NOW variable (set in the %post section, above). The options passed to the container at runtime are printed as a single string ($*) and then they are passed to echo via a quoted array ($@) which ensures that all of the arguments are properly parsed by the executed command. The exec preceding the final echo command replaces the current entry in the process table (which originally was the call

to SingularityCE). Thus, the runscript shell process ceases to exist, and only the process running within the container remains.

Running the container built using this def file will yield the following:

```
$ ./my_container.sif
Container was created Thu Dec  6 20:01:56 UTC 2018
Arguments received:

$ ./my_container.sif this that and the other
Container was created Thu Dec  6 20:01:56 UTC 2018
Arguments received: this that and the other
this that and the other
```

The `%runscript` scriptlet will run under `sh` or `bash` by default. You can change the shell or interpreter that the test runs under by using a custom hashbang (`#!`) as the first line in your `%runscript` section:

```
%runscript
   #!/bin/zsh

   echo "$(readlink /proc/$$/exe) is our shell"
```

Just like in the *`%test` section*, the `#!/bin/zsh` means that the runscript code will be run by the zsh shell installed at `/bin/zsh`. The `echo` statement given above will display the shell that is running the script, confirming that this works.

And just like in the *`%test` section*, a custom hashbang runs the specified shell from the container filesystem, not the host. Therefore, `zsh` must be installed in the *container*, and since `zsh` is not built into the base Ubuntu image, it would have to be installed as part of the `%post` section for this `%runscript` code to work properly.

%labels

The `%labels` section is used to add metadata to the file `/.singularity.d/labels.json` within your container. The general format is a name-value pair.

Consider the `%labels` section from the example definition file above:

```
%labels
    Author myuser@example.com
    Version v0.0.1
    MyLabel Hello World
```

Note that labels are key-value pairs. To define a new label, add a new line of text to the `%labels` section. The portion of text up to the first space will be taken as the label's name, and the portion following it will be taken as the label's value.

In the previous example, the first label name is `Author` with a value of `myuser@example.com`. The second label name is `Version` with a value of `v0.0.1`. Finally, the third label name is `MyLabel` with a value of `Hello World`.

You can inspect the available labels on your image by running the following command:

```
$ singularity inspect my_container.sif

{
  "Author": "myuser@example.com",
  "Version": "v0.0.1",
  "MyLabel": "Hello World",
```

(continues on next page)

(continued from previous page)

```
  "org.label-schema.build-date": "Thursday_6_December_2018_20:1:56_UTC",
  "org.label-schema.schema-version": "1.0",
  "org.label-schema.usage": "/.singularity.d/runscript.help",
  "org.label-schema.usage.singularity.deffile.bootstrap": "library",
  "org.label-schema.usage.singularity.deffile.from": "ubuntu:18.04",
  "org.label-schema.usage.singularity.runscript.help": "/.singularity.d/runscript.help",
  "org.label-schema.usage.singularity.version": "3.0.1"
}
```

As you can see from this output, some labels are generated automatically from the build process. You can read more about labels and metadata *here*.

%help

Any text in the `%help` section is transcribed into a dedicated metadata file in the container during the build process. This text can then be displayed using the `run-help` command.

Consider the `%help` section from the example definition file above:

```
%help
    This is a demo container used to illustrate a def file that uses all
    supported sections.
```

After building the help can be displayed like so:

```
$ singularity run-help my_container.sif
    This is a demo container used to illustrate a def file that uses all
    supported sections.
```

2.2.4 Templating: How to Pass Values into Definition Files

Starting with version 4.0, SingularityCE definition files support *templating*: definition files can include placeholders for values that will be passed at build time, either using command-line options or by specifying a file that contains the relevant values.

Basics

To use templating, include a `{{ placeholder }}` at the point in your definition file where you'd like the passed-in value to go. For example:

```
Bootstrap: library
From: ubuntu:22.04
Stage: build

%runscript
    echo {{ some_text }}
```

When building a container from this definition file, a concrete value for `{{ some_text }}` can be passed via the `--build-arg` flag to the `build` command. This flag accepts a `varname=value` pair, as shown here:

```
$ sudo singularity build --build-arg some_text="Hello world" ./my_container.sif ./my_
↪container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
Hello world
```

Alternatively, the `varname=value` assignments can be placed in a file, and the path to that file specified using the `--build-arg-file` flag to the `build` command, as shown here:

```
$ cat << EOF > ./my_args_file.txt
some_text="Hello again, world"
EOF

$ sudo singularity build -F --build-arg-file ./my_args_file.txt ./my_container.sif ./my_
↪container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
Hello again, world
```

Working with multiple variables

A single definition file can use multiple different templating variables, use a single variable more than once, and use variables in different sections of the definition file, as shown here:

```
Bootstrap: library
From: ubuntu:22.04
Stage: build

%setup
    echo {{ file_contents }} > /tmp/test_file

%environment
    export CUSTOM_VAR_ONE={{ var_value1 }}
    export CUSTOM_VAR_TWO={{ var_value2 }}

%runscript
    echo "file contents:" `cat /tmp/test_file`
    echo "--> this should, if I'm not mistaken, equal:" {{ file_contents }}
```

(continues on next page)

(continued from previous page)

```
    echo ""
    echo "env var: ${CUSTOM_VAR_ONE}"
    echo "--> this should, if I'm not mistaken, equal:" {{ var_value1 }}
    echo ""
    echo "env var: ${CUSTOM_VAR_TWO}"
    echo "--> this should, if I'm not mistaken, equal:" {{ var_value2 }}
    echo ""
    echo "and finally, here's some text:" {{ some_text}}
```

To use this definition file, one can pass values for all the different variables using `--build-arg` flags:

```
$ sudo singularity build -F --build-arg file_contents="'I am in a file'" --build-arg var_
↪value1="'I am in an env var'" --build-arg var_value2="'I am also in an env var'" --
↪build-arg some_text="'I am just some text'" ./my_container.sif ./my_container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Running setup scriptlet
+ echo 'I am in a file'
INFO:    Adding environment to container
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
file contents: I am in a file
--> this should, if I'm not mistaken, equal: I am in a file

env var: I am in an env var
--> this should, if I'm not mistaken, equal: I am in an env var

env var: I am also in an env var
--> this should, if I'm not mistaken, equal: I am also in an env var

and finally, here's some text: I am just some text
```

Or one can place the values for the different values in a file, and pass the path to that file using the `--build-arg-file` flag:

```
$ cat << EOF > ./my_args_file.txt
file_contents="I am in a file"
var_value1="I am in an env var"
var_value2="I am also in an env var"
some_text="I am just some text"
EOF

$ sudo singularity build -F --build-arg-file ./my_args_file.txt ./my_container.sif ./my_
↪container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
```

(continues on next page)

(continued from previous page)

```
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Running setup scriptlet
+ echo 'I am in a file'
INFO:    Adding environment to container
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
file contents: I am in a file
--> this should, if I'm not mistaken, equal: I am in a file

env var: I am in an env var
--> this should, if I'm not mistaken, equal: I am in an env var

env var: I am also in an env var
--> this should, if I'm not mistaken, equal: I am also in an env var

and finally, here's some text: I am just some text
```

Or one can use a combination of both strategies:

```
$ cat << EOF > ./my_args_file.txt
var_value1="I am in an env var"
some_text="I am just some text"
EOF

$ sudo singularity build -F --build-arg file_contents="'I am in a file'" --build-arg var_
↪value2="'I am also in an env var'" --build-arg-file ./my_args_file.txt ./my_container.
↪sif ./my_container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Running setup scriptlet
+ echo 'I am in a file'
INFO:    Adding environment to container
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
file contents: I am in a file
--> this should, if I'm not mistaken, equal: I am in a file

env var: I am in an env var
--> this should, if I'm not mistaken, equal: I am in an env var

env var: I am also in an env var
--> this should, if I'm not mistaken, equal: I am also in an env var

and finally, here's some text: I am just some text
```

Precedence among multiple value sources

In the event that an argument is passed via `--build-arg` more than once, the last occurrence will take precedence:

```
$ sudo singularity build -F --build-arg file_contents="'I am in a file (1st time)'" --
↪build-arg var_value2="'I am also in an env var'" --build-arg file_contents="'I am in a␣
↪file (2nd time)'" --build-arg-file ./my_args_file.txt ./my_container.sif ./my_
↪container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Running setup scriptlet
+ echo 'I am in a file (2nd time)'
INFO:    Adding environment to container
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
file contents: I am in a file (2nd time)
--> this should, if I'm not mistaken, equal: I am in a file (2nd time)

env var: I am in an env var
--> this should, if I'm not mistaken, equal: I am in an env var

env var: I am also in an env var
--> this should, if I'm not mistaken, equal: I am also in an env var

and finally, here's some text: I am just some text
```

In the event that a variable is defined both in the file passed to `--build-arg-file` and via the command line using `--build-arg` flag, the value passed via the command line will take precedence:

```
$ cat << EOF > ./my_args_file.txt
var_value1="I am in an env var"
var_value2="I am also in an env var (from build arg file)"
some_text="I am just some text"
EOF

$ sudo singularity build -F --build-arg file_contents="'I am in a file'" --build-arg var_
↪value2="'I am also in an env var (from command line)'" --build-arg-file ./my_args_file.
↪txt ./my_container.sif ./my_container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
INFO:    Running setup scriptlet
+ echo 'I am in a file'
INFO:    Adding environment to container
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif
```

(continues on next page)

(continued from previous page)

```
$ singularity run ./my_container.sif
file contents: I am in a file
--> this should, if I'm not mistaken, equal: I am in a file

env var: I am in an env var
--> this should, if I'm not mistaken, equal: I am in an env var

env var: I am also in an env var (from command line)
--> this should, if I'm not mistaken, equal: I am also in an env var (from command line)

and finally, here's some text: I am just some text
```

Default variable values: the %arguments section

If a definition file contains a variable placeholder and no value for that variable is provided (via either `--build-arg` or `--build-arg-file`), SingularityCE will generate an error:

```
Bootstrap: library
From: ubuntu:22.04
Stage: build

%runscript
   echo "Here is some text:" {{ some_text }}
   echo "And here is some more text:" {{ some_more_text }}
```

```
$ sudo singularity build -F --build-arg some_more_text="more more more" ./my_container.
↪sif ./my_container.def
FATAL:   Unable to build from ./my_container.def: build var some_text is not defined␣
↪through either --build-arg (--build-arg-file) or 'arguments' section
```

However, definition files can provide default values for some or all variables using the `%arguments` section, as shown here:

```
Bootstrap: library
From: ubuntu:22.04
Stage: build

%runscript
   echo "Here is some text:" {{ some_text }}
   echo "And here is some more text:" {{ some_more_text }}

%arguments
   some_text="Some default text"
   some_more_text="Some more default text"
```

```
$ sudo singularity build -F --build-arg some_more_text="more more more" ./my_container.
↪sif ./my_container.def
INFO:    Starting build...
INFO:    Using cached image
INFO:    Verifying bootstrap image /root/.singularity/cache/library/sha256.
↪7a63c14842a5c9b9c0567c1530af87afbb82187444ea45fd7473726ca31a598b
```

(continues on next page)

(continued from previous page)

```
INFO:    Adding runscript
INFO:    Creating SIF file...
INFO:    Build complete: ./my_container.sif

$ singularity run ./my_container.sif
Here is some text: Some default text
And here is some more text: more more more
```

Note that in the event of a variable having both a default value *and* a value explicitly set via `--build-arg` or `--build-arg-file`, the explicitly-set value will take precedence (as the example above shows).

2.2.5 Multi-Stage Builds

Starting with version 3.2, SingularityCE supports multi-stage builds, where one environment can be used for compilation, and the resulting binary can then be copied into a different final environment. One of the important advantages of this approach is that it allows for a slimmer final image that does not require the entire development stack.

```
Bootstrap: docker
From: golang:1.12.3-alpine3.9
Stage: devel

%post
  # prep environment
  export PATH="/go/bin:/usr/local/go/bin:$PATH"
  export HOME="/root"
  cd /root

  # insert source code, could also be copied from the host with %files
  cat << EOF > hello.go
  package main
  import "fmt"

  func main() {
    fmt.Printf("Hello World!\n")
  }
EOF

  go build -o hello hello.go


# Install binary into the final image
Bootstrap: library
From: alpine:3.9
Stage: final

# install binary from stage one
%files from devel
  /root/hello /bin/hello

%runscript
  /bin/hello
```

The names of stages (assigned using the `Stage` keyword) are arbitrary. Each of these sections will be executed in the same order as described for a single stage build, except that the files from the previous stage are copied before the `%setup` section of the next stage is carried out. Files can only be copied from stages declared before the current stage in the definition. E.g., the `devel` stage in the above definition cannot copy files from the `final` stage, but the `final` stage can copy files from the `devel` stage.

2.2.6 SCIF Apps

SCIF is a standard for encapsulating multiple apps into a container. A container with SCIF apps has multiple entry points, and it is easy to choose which one you want to run. Each entry point can carry out a different task, with its own environment, metadata, etc., without the need for a collection of different containers.

SingularityCE implements SCIF, and you can read more about how to use it below.

SCIF is not specific to SingularityCE. To learn more, take a look at the project's site at https://sci-f.github.io/, which includes extended tutorials, a detailed specification of the SCIF standard, and other information.

SCIF %app* sections

SCIF apps within a SingularityCE container are created using `%app*` sections in a definition file. These `%app*` sections, which will impact the way the container runs a specific `--app`, can exist alongside any of the primary sections (i.e. `%post`,``%runscript``, `%environment`, etc.). As with other sections, the ordering of the `%app*` sections isn't important.

The following runscript demonstrates how to build 2 different apps into the same container using SCIF modules:

```
Bootstrap: docker
From: ubuntu

%environment
    GLOBAL=variables
    AVAILABLE="to all apps"

##############################
# foo
##############################

%apprun foo
    exec echo "RUNNING FOO"

%applabels foo
    BESTAPP FOO

%appinstall foo
    touch foo.exec

%appenv foo
    SOFTWARE=foo
    export SOFTWARE

%apphelp foo
    This is the help for foo.

##############################
```

(continues on next page)

(continued from previous page)

```
# bar
##############################

%apphelp bar
    This is the help for bar.

%applabels bar
    BESTAPP BAR

%appinstall bar
    touch bar.exec

%apprun bar
    exec echo "RUNNING BAR"

%appenv bar
    SOFTWARE=bar
    export SOFTWARE
```

An `%appenv` section is the app-specific equivalent of `%environment`.

Similarly, `%appinstall` is like `%post` but for a particular app. Note that just like the general `%post` section, `%appinstall` sections run at build time. Thus, when building a container from a definition file containing `%appinstall` sections, the content of all of these sections will be executed—even if later on, the user ends up running only some of the apps defined in the file and not others. This is why the SCIF Standard indicates that files & directories that are app-specific, and are potentially mutually-exclusive with the files & directories of other apps, be placed under the app-specific `/scif/apps/<app-name>` directory to avoid conflicts between different apps.

Installing apps into modules using the `%app*` sections enables the `--app` option, allowing commands like the following:

```
% singularity run --app foo my_container.sif
RUNNING FOO
```

This runs a specific app, `foo`, from the multi-app container we built.

The same environment variable, `$SOFTWARE` is defined for both apps in the def file above. You can execute the following command to search the list of active environment variables and `grep` to determine if the variable changes depending on the app we specify:

```
$ singularity exec --app foo my_container.sif env | grep SOFTWARE
SOFTWARE=foo

$ singularity exec --app bar my_container.sif env | grep SOFTWARE
SOFTWARE=bar
```

2.2.7 Best Practices for Writing Definition Files

When crafting your definition file, it is best to consider the following:

1. Always install packages, programs, data, and files into operating system locations (e.g. not `/home`, `/tmp` , or any other directories that might get commonly bind mounted to host directories).
2. Document your container. If your runscript doesn't supply help, write a `%help` or `%apphelp` section. A good container tells the user how to interact with it.
3. If you require any special environment variables to be defined, add them to the `%environment` and `%appenv` sections of the definition file.
4. Files should always be owned by a system account (UID lower than 500).
5. Ensure that sensitive files like `/etc/passwd`, `/etc/group`, and `/etc/shadow` do not contain secrets.
6. Build production containers from a definition file instead of a sandbox that has been manually changed. This ensures maximal reproducibility, and mitigates the possibility of your production container being a "black box."

2.3 Build Environment

2.3.1 Overview

You may wish to customize your build environment by doing things such as specifying a custom cache directory for images, or sending your Docker Credentials to the registry endpoint. In this section, we will discuss these and other topics related to the build environment.

2.3.2 Cache Folders

SingularityCE will cache SIF container images generated from remote sources, and any OCI/docker layers used to create them. The cache is created at `$HOME/.singularity/cache` by default. The location of the cache can be changed by setting the `SINGULARITY_CACHEDIR` environment variable.

When you run builds as root using `sudo`, images will be cached in root's home directory at `/root`, rather than your user's home directory. If you have set the `SINGULARITY_CACHEDIR` environment variable, you may use `sudo`'s `-E` option to pass the value of `SINGULARITY_CACHEDIR` through to the root user's environment. This allows you to control where images will be cached even when running builds under `sudo`.

```
$ export SINGULARITY_CACHEDIR=/tmp/user/temporary-cache

# Running a build under your user account
$ singularity build --fakeroot myimage.sif mydef.def

# Running a build with sudo, must use -E to pass env var
$ sudo -E singularity build myimage.sif mydef.def
```

If you change the value of `SINGULARITY_CACHEDIR` be sure to choose a location that is:

- Unique to you. Permissions are set on the cache so that private images cached for one user are not exposed to another. This means that SingularityCE cache directories cannot be shared across users.
- Located on a filesystem with sufficient space for the number and size of container images you anticipate using.
- Located on a filesystem that supports atomic rename, if possible.

Warning: If you are not certain that your `$HOME` or `SINGULARITY_CACHEDIR` filesystems support atomic rename, do not run SingularityCE in parallel using remote container URLs. Instead, use `singularity pull` to create a local SIF image, and then run this SIF image in a parallel step. Alternatively, you may use the `--disable-cache` option, but this will result in each SingularityCE instance independently fetching the container from the remote source, into a temporary location.

Inside the cache location you will find separate directories for the different kinds of data that are cached:

```
$HOME/.singularity/cache/blob
$HOME/.singularity/cache/library
$HOME/.singularity/cache/net
$HOME/.singularity/cache/oci-sif
$HOME/.singularity/cache/oci-tmp
$HOME/.singularity/cache/oras
$HOME/.singularity/cache/shub
```

You can safely delete these directories, or content within them. SingularityCE will re-create any directories and data that are needed in future runs.

You should not add any additional files, or modify files in the cache, as this may cause checksum / integrity errors when you run or build containers. If you experience problems, use `singularity cache clean` to reset the cache to a clean, empty state.

BoltDB Corruption Errors

The library that SingularityCE uses to retrieve and cache Docker/OCI layers keeps track of them using a single-file database. If your home directory is on a network filesystem which experiences interruptions, or you run out of storage, it is possible for this database to become inconsistent.

If you observe error messages that mention *github.com/etcd-io/bbolt* when trying to run SingularityCE, then you should remove the database file:

```
rm ~/.local/share/containers/cache/blob-info-cache-v1.boltdb
```

2.3.3 Cache commands

The `cache` command for SingularityCE allows you to view and clean up your cache, without needing to manually inspect the cache directories.

Note: If you have built images as root, directly or via `sudo`, the default cache location for those builds is `/root/.singularity`. You will need to use `sudo` when running `cache clean` or `cache list` to manage these cache entries.

Listing the Cache

To view a summary of cache usage, use `singularity cache list`:

```
$ singularity cache list
There are 5 container file(s) using 74.80 MiB and 18 oci blob file(s) using 71.70 MiB of␣
↪space
Total space used: 146.50 MiB
```

To view more detailed information, use `singularity cache list -v`:

```
$ singularity cache list -v
NAME                     DATE CREATED           SIZE             TYPE
07a18d51e256ea8c9e8de0   2023-08-14 16:09:04    1.75 KiB         blob
278d875d73f02153bf7ed2   2023-08-14 16:09:03    0.15 KiB         blob
332c15a4bec38b7947aec0   2023-08-14 16:09:03    0.13 KiB         blob
553345aafebc934b169982   2023-08-14 12:14:24    0.95 KiB         blob
7176c5ea9d28ae84d6accb   2023-08-14 16:09:03    0.20 KiB         blob
7264a8db6415046d36d16b   2023-08-11 17:13:59    3.24 MiB         blob
913cf3a39d377faf89ed38   2023-08-11 17:13:59    0.57 KiB         blob
9fda8d8052c61740409c4b   2023-08-14 16:09:04    3.18 MiB         blob
a1d08a2769560809bf03ba   2023-08-14 16:09:04    0.20 KiB         blob
b3283fa64ecd626e391440   2023-08-14 12:14:39    0.99 KiB         blob
b9e0aa7145707602cfc584   2023-08-14 16:09:03    0.12 KiB         blob
c5c5fda71656f28e49ac9c   2023-08-11 17:13:53    1.60 KiB         blob
cc82f5d421a1914e2ce2a8   2023-08-14 12:14:39    0.40 KiB         blob
cf4e5bc0709f07284518b2   2023-08-11 17:13:59    0.40 KiB         blob
deb9cd9f829fea30353f8c   2023-08-14 12:14:39    65.27 MiB        blob
eb9556ecd24f1fa496f2f7   2023-08-14 16:09:03    0.15 KiB         blob
f1dc9184bcff6fbdfd18dc   2023-08-14 16:09:04    0.15 KiB         blob
fd4ed8f3240239c3dde6dc   2023-08-14 16:09:04    2.59 KiB         blob
sha256.9a6ee1f8fdecb21   2023-08-31 09:54:44    2.65 MiB         library
sha256:07a18d51e256ea8   2023-08-14 16:09:05    3.11 MiB         oci-tmp
sha256:c5c5fda71656f28   2023-08-16 14:00:27    3.19 MiB         oci-tmp
sha256:553345aafebc934   2023-08-14 12:14:55    62.68 MiB        oci-sif
sha256:c5c5fda71656f28   2023-08-15 09:02:11    3.18 MiB         oci-sif

There are 5 container file(s) using 74.80 MiB and 18 oci blob file(s) using 71.70 MiB of␣
↪space
Total space used: 146.50 MiB
```

All cache entries are named using a content hash, so that identical layers or images that are pulled from different URIs do not result in duplication within the cache.

You can limit the cache list to a specific cache type with the `--type` / `-t` option. The cache types are:

- **blob**: Configuration and filesystem layers for OCI containers that have been retrieved from a registry or other source.
- **library**: SIF images retrieved from a `library://` source.
- **net**: SIF, squashfs, and extfs images retrieved from `http/https` URIs.
- **oci-sif**: OCI-SIF images created from OCI blobs. These are cached to avoid multiple conversions when a container is run repeatedly from an OCI URI (`singularity run --oci docker://alpine`).

- **oci-tmp**: SIF images created from OCI blobs. These are cached to avoid multiple conversions when a container is run repeatedly from an OCI URI (`singularity run docker://alpine`).
- **oras**: SIF images retrieved from an OCI registry via the `oras://` protocol.
- **shub**: SIF, squashfs, and extfs images retrieved from a `shub://` source.

Cleaning the Cache

To reclaim space used by the SingularityCE cache, use `singularity cache clean`.

By default, `singularity cache clean` will remove all cache entries, after asking you to confirm:

```
$ singularity cache clean
This will delete everything in your cache (containers from all sources and OCI blobs).
Hint: You can see exactly what would be deleted by canceling and using the --dry-run␣
↪option.
Do you want to continue? [N/y] n
```

Use the `--dry-run` / `-n` option to see the files that would be deleted, or the `--force` / `-f` option to clean without asking for confirmation.

If you want to leave your most recent cached images in place, but remove images that were cached longer ago, you can use the `--days` / `-d` option. E.g. to clean cache entries older than 30 days:

```
$ singularity cache clean --days 30
```

To remove only a specific kind of cache entry, e.g. only library images, use the `type` / `-T` option:

```
$ singularity cache clean --type library
```

2.3.4 Temporary Folders

When building a container, or pulling/running a SingularityCE container from a Docker/OCI source, a temporary working space is required. The container is constructed in this temporary space before being packaged into a SingularityCE SIF image. Temporary space is also used when running containers in unprivileged mode, and when performing certain operations on filesystems that do not fully support `--fakeroot`.

The location for temporary directories defaults to `/tmp`. However, SingularityCE will respect the environment variable `TMPDIR`, and both of these locations can be overridden by setting the environment variable `SINGULARITY_TMPDIR`.

The temporary directory used during a build must be on a filesystem that has enough space to hold the entire container image, uncompressed, including any temporary files that are created and later removed in the course of the build. You may therefore need to set `SINGULARITY_TMPDIR` when building a large container on a system which has a small `/tmp` filesystem.

Remember to use `-E` option to pass the value of `SINGULARITY_TMPDIR` through to root's environment when executing the `build` command with `sudo`.

Warning: Many modern Linux distributions use an in-memory `tmpfs` filesystem for `/tmp` when installed on a computer with a sufficient amount of RAM. This may limit the size of container you can build, as temporary directories under `/tmp` share RAM with runniing programs etc. A `tmpfs` also uses default mount options that can interfere with some container builds.

If you experience problems, set `SINGULARITY_TMPDIR` to a disk location, or disable the `tmpfs /tmp` mount on your system .

2.3.5 Encrypted Containers

Starting with SingularityCE 3.4.0, it is possible to build and run encrypted containers. The containers are decrypted at runtime entirely in kernel space, meaning that no intermediate decrypted data is ever written to disk. See *encrypted containers* for more details.

2.3.6 Environment Variables

You can set environment variables on the host to influence the behaviour of a build. Note that environment variables are not passed into the build itself, and cannot be accessed in the `%post` section of a definition file. To pass values into a build, use the *templating / build-args support* introduced in SingularityCE 4.0.

Environment variables that control a build are generally associated with an equivalent CLI option. The order of precendence is:

1. If a flag is represented by both a CLI option and an environment variable, and both are set, the CLI option will take precedence. This is true for all environment variables with the exception of `SINGULARITY_BIND` and `SINGULARITY_BINDPATH`, which are combined with the `--bind` option / argument pair, if both are present.
2. Environment variables will override default values of CLI options that have not been explicitly set in the command line.
3. Any default values for CLI options that have not been overridden on the command line, or by corresponding environment variables, will then take effect.

Defaults

The following variables have defaults that can be overridden by assigning your own values to the corresponding environment variables at runtime:

Docker

`SINGULARITY_DOCKER_LOGIN`
Set this to login to a Docker Repository interactively.

`SINGULARITY_DOCKER_USERNAME`
Your Docker username.

`SINGULARITY_DOCKER_PASSWORD`
Your Docker password.

`RUNSCRIPT_COMMAND`
Is not obtained from the environment, but is a hard coded default ("/bin/bash"). This is the fallback command used in the case that the docker image does not have a CMD or ENTRYPOINT. `TAG` Is the default tag, `latest`.

`SINGULARITY_NOHTTPS`
This is relevant if you want to use a registry that doesn't support https. A typical use-case for this variable is when using local registry, running on the same machine as SingularityCE itself.

Library

`SINGULARITY_BUILDER`
Used to specify the remote builder service URL. The default value is Sylabs' remote builder.

`SINGULARITY_LIBRARY`
Used to specify the library to pull from.
Default is set to Sylabs' Cloud Library.

`SINGULARITY_REMOTE`
Used to build an image remotely. (Importantly, such remote builds do not require root access on the local machine.) The default is false.

Encryption

`SINGULARITY_ENCRYPTION_PASSPHRASE`
Used to pass a plaintext passphrase to be used to encrypt a container file system (in conjunction with the `--encrypt` flag). The default is empty.

`SINGULARITY_ENCRYPTION_PEM_PATH`
Used to specify the location of a public key to use for container encryption (in conjunction with the `--encrypt` flag). The default is empty.

2.4 Fakeroot feature

2.4.1 Overview

The fakeroot feature (commonly referred as rootless mode) allows an unprivileged user to run a container as a **"fake root"** user by leveraging user namespace UID/GID mapping.

Note: This feature requires a Linux kernel >= 3.8, but the recommended version is >= 3.18

A **"fake root"** user has almost the same administrative rights as root but only **inside the container** and the **requested namespaces**, which means that this user:

- can set different user/group ownership for files or directories they own
- can change user/group identity with su/sudo commands
- has full privileges inside the requested namespaces (network, ipc, uts)

2.4.2 Restrictions/security

Filesystem

A **"fake root"** user can't access or modify files and directories for which they don't already have access or rights on the host filesystem, so a **"fake root"** user won't be able to access root-only host files like `/etc/shadow` or the host `/root` directory.

Additionally, all files or directories created by the **"fake root"** user are owned by `root:root` inside container but as `user:group` outside of the container. Let's consider the following example, in this case "user" is authorized to use the fakeroot feature and can use 65536 UIDs starting at 131072 (same thing for GIDs).

UID inside container	UID outside container
0 (root)	1000 (user)
1 (daemon)	131072 (non-existent)
2 (bin)	131073 (non-existent)
...	...
65536	196607

Which means if the **"fake root"** user creates a file under a `bin` user in the container, this file will be owned by `131073:131073` outside of container. The responsibility relies on the administrator to ensure that there is no overlap with the current user's UID/GID on the system.

Network

Restrictions are also applied to networking, if `singularity` is executed without the `--net` flag, the **"fake root"** user won't be able to use `ping` or bind a container service to a port below 1024.

With `--net` the **"fake root"** user has full privileges in a dedicated container network. Inside the container network they can bind on privileged ports below 1024, use ping, manage firewall rules, listen to traffic, etc. Anything done in this dedicated network won't affect the host network.

Note: Of course an unprivileged user could not map host ports below than 1024 by using: `--network-args="portmap=80:80/tcp"`

Warning: For unprivileged installation of SingularityCE or if `allow setuid = no` is set in `singularity.conf` users won't be able to use a `fakeroot` network.

2.4.3 Requirements / Configuration

Fakeroot depends on user mappings set in `/etc/subuid` and group mappings in `/etc/subgid`, so your username needs to be listed in those files with a valid mapping (see the admin-guide for details), if you can't edit the files ask an administrator.

In SingularityCE `3.5` a `singularity config fakeroot` command has been added to allow configuration of the `/etc/subuid` and `/etc/subgid` mappings from the SingularityCE command line. You must be a root user or run with `sudo` to use `config fakeroot`, as the mapping files are security sensitive. See the admin-guide for more details.

2.4.4 Usage

If your user account is configured with valid `subuid` and `subgid` mappings you work as a fake root user inside a container by using the `--fakeroot` or `-f` option.

The `--fakeroot` option is available with the following singularity commands:

- `shell`
- `exec`
- `run`
- `instance start`
- `build`

Build

With fakeroot an unprivileged user can now build an image from a definition file with few restrictions. Some bootstrap methods that require creation of block devices (like `/dev/null`) may not always work correctly with **"fake root"**, SingularityCE uses seccomp filters to give programs the illusion that block device creation succeeded. This appears to work with `yum` bootstraps and *may* work with other bootstrap methods, although `debootstrap` is known to not work.

Examples

Build from a definition file:

```
singularity build --fakeroot /tmp/test.sif /tmp/test.def
```

Ping from container:

```
singularity exec --fakeroot --net docker://alpine ping -c1 8.8.8.8
```

HTTP server:

```
singularity run --fakeroot --net --network-args="portmap=8080:80/tcp" -w docker://nginx
```

CHAPTER

THREE

CONTAINER SIGNING & ENCRYPTION

SingularityCE allows containers to be signed using a PGP key. The signature travels with the container image, allowing you to verify that the image is unmodified at any time. Encryption of containers using LUKS2 is also supported. Encrypted containers can be run without decrypting them to disk first.

3.1 Signing and Verifying Containers

SingularityCE's SIF images can be signed, and subsequently verified, so that a user can be confident that an image they have obtained is a bit-for-bit reproduction of the original container as the author intended it. The signature, over the metadata and content of the container, is created using a private key, and directly added to the SIF file. This means that a signed container carries it's signature with it, avoiding the need for extra infrastructure to distribute signatures to end users of the container.

A user verifies the container has not been modified since signing using a public key or certificate. By default, SingularityCE uses PGP keys to sign and verify containers. Since 3.11, signing and verifying containers with X.509 key material / certificates is also supported.

PGP Public keys for verification can be distributed manually, or can be upload to and automatically retrieved from Singularity Container Services (SCS) or a Singularity Enterprise installation. Optionally, an HKP keyserver can also be configured.

As well as indicating a container has not been modified, a valid signature may be used to indicate a container has undergone testing or review, and is approved for execution. Multiple signatures can be added to a container, to document its progress through an approval process. SingularityCE's Execution Control List (ECL) feature can be enable by administrators of privileged installations to restrict execution of containers based on their signatures (see the admin guide for more information).

Note: Due to a change in signature format, containers signed by 3.6.0 and later cannot be verified by older versions of SingularityCE.

To verify containers signed with older versions of SingularityCE using 3.6.0 and above, the `--legacy-insecure` flag must be provided to the `singularity verify` command.

3.1.1 Verifying containers from the Container Library

The `verify` command will check that a SIF container image has been signed using a PGP key, and is unchanged since it was signed.

If you are using a container image that was pulled from the SCS container library, then it is likely that it was signed with a PGP key that has been submitted to the SCS keystore. SingularityCE is able to automatically retrieve the public key to perform verification.

```
$ singularity pull library://alpine:latest

$ singularity verify alpine_latest.sif

Container is signed by 1 key(s):

Verifying partition: FS:
8883491F4268F173C6E5DC49EDECE4F3F38D871E
[REMOTE]  Sylabs Admin <support@sylabs.io>
[OK]      Data integrity verified

INFO:    Container verified: alpine_latest.sif
```

In this example you can see that **Sylabs Admin** has signed the container.

Note: `singularity verify` will only run against a local SIF file. You must `pull` an image from a `library://` source before you can verify it.

3.1.2 Signing your own containers

Generating and managing PGP keys

To sign your own containers you first need to generate one or more keys. In order to submit them to the SCS keystore, you will also need to login to SCS with a token:

1. Go to: https://cloud.sylabs.io/
2. Click "Sign In" and follow the sign in steps.
3. Click on your login id (same and updated button as the Sign in one).
4. Select "Access Tokens" from the drop down menu.
5. Enter a name for your new access token, such as "test token"
6. Click the "Create a New Access Token" button.
7. Click "Copy token to Clipboard" from the "New API Token" page.
8. Run `singularity remote login` and paste the access token at the prompt.

If you attempt to sign a container before you have generated any keys, SingularityCE will guide you through the interactive process of creating a new key. Or you can use the `newpair` subcommand in the `key` command group like so:.

```
$ singularity key newpair

Enter your name (e.g., John Doe) : Joe User
Enter your email address (e.g., john.doe@example.com) : myuser@example.com
Enter optional comment (e.g., development keys) : demo
Enter a passphrase :
Retype your passphrase :
Would you like to push it to the keystore? [Y,n] Y
Generating Entity and OpenPGP Key Pair... done
Key successfully pushed to: https://keys.sylabs.io
```

Note that I chose `Y` when asked if I wanted to push my key to the keystore. This will push my public key to whichever keystore has been configured by the `singularity remote` command, so that it can be retrieved by other users running `singularity verify`. If you do not wish to push your public key, say `n` during the `newpair` process.

The `list` subcommand will show you all of the keys you have created or saved locally.`

```
$ singularity key list

Public key listing (/home/dave/.singularity/sypgp/pgp-public):

0)  User:              Joe User (demo) <myuser@example.com>
    Creation time:     2019-11-15 09:54:54 -0600 CST
    Fingerprint:       E5F780B2C22F59DF748524B435C3844412EE233B
    Length (in bits):  4096
```

If you chose not to push your key to the keystore during the `newpair` process, but later wish to, you can push it to a keystore configured using `singularity remote` like so:

```
$ singularity key push E5F780B2C22F59DF748524B435C3844412EE233B

public key `E5F780B2C22F59DF748524B435C3844412EE233B` pushed to server successfully
```

If you delete your local public PGP key, you can always locate and download it again like so.

```
$ singularity key search Trudgian

Showing 1 results

KEY ID    BITS  NAME/EMAIL
12EE233B  4096  Joe User (demo) <myuser@example.com>

$ singularity key pull 12EE233B

1 key(s) added to keyring of trust /home/dave/.singularity/sypgp/pgp-public
```

But note that this only restores the *public* key (used for verifying) to your local machine and does not restore the *private* key (used for signing).

Searching for keys

SingularityCE allows you to search the keystore for public keys. You can search for names, emails, and fingerprints (key IDs). When searching for a fingerprint, you need to use `0x` before the fingerprint, check the example:

```
# search for key ID:
$ singularity key search 0x8883491F4268F173C6E5DC49EDECE4F3F38D871E

# search for the sort ID:
$ singularity key search 0xF38D871E

# search for user:
$ singularity key search Godlove

# search for email:
$ singularity key search @gmail.com
```

Signing and validating your own containers

Now that you have a key generated, you can use it to sign images like so:

```
$ singularity sign my_container.sif

Signing image: my_container.sif
Enter key passphrase :
Signature created and applied to my_container.sif
```

Because your public PGP key is saved locally you can verify the image without needing to contact the Keystore.

```
$ singularity verify my_container.sif
Verifying image: my_container.sif
[LOCAL]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[LOCAL]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
1   |1       |NONE    |Def.FILE
2   |1       |NONE    |JSON.Generic
3   |1       |NONE    |FS
Container verified: my_container.sif
```

If you've pushed your key to the Keystore you can also verify this image in the absence of a local public key. To demonstrate this, first `remove` your local public key, and then try to use the `verify` command again.

```
$ singularity key remove E5F780B2C22F59DF748524B435C3844412EE233B

$ singularity verify my_container.sif
Verifying image: my_container.sif
[REMOTE]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[REMOTE]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
```

(continues on next page)

(continued from previous page)

```
1   |1       |NONE    |Def.FILE
2   |1       |NONE    |JSON.Generic
3   |1       |NONE    |FS
Container verified: my_container.sif
```

Note that the `[REMOTE]` message shows the key used for verification was obtained from the keystore, and is not present on your local computer. You can retrieve it, so that you can verify even if you are offline with `singularity key pull`

```
$ singularity key pull E5F780B2C22F59DF748524B435C3844412EE233B

1 key(s) added to keyring of trust /home/dave/.singularity/sypgp/pgp-public
```

Advanced Signing - SIF IDs and Groups

As well as the default behaviour, which signs all objects, fine-grained control of signing is possible.

If you `sif list` a SIF file you will see it is comprised of a number of objects. Each object has an `ID`, and belongs to a `GROUP`.

```
$ singularity sif list my_container.sif

Container id: e455d2ae-7f0b-4c79-b3ef-315a4913d76a
Created on:   2019-11-15 10:11:58 -0600 CST
Modified on:  2019-11-15 10:11:58 -0600 CST
-----------------------------------------------------
Descriptor list:
ID   |GROUP   |LINK    |SIF POSITION (start-end)  |TYPE
------------------------------------------------------------------------------
1    |1       |NONE    |32768-32800               |Def.FILE
2    |1       |NONE    |36864-36961               |JSON.Generic
3    |1       |NONE    |40960-25890816            |FS (Squashfs/*System/amd64)
```

Note: The `singularity sif` commands will only run against a local SIF file. You must `pull` an image from a `library://` source before you can examine it.

I can choose to sign and verify a specific object with the `--sif-id` option to `sign` and `verify`.

```
$ singularity sign --sif-id 1 my_container.sif
Signing image: my_container.sif
Enter key passphrase :
Signature created and applied to my_container.sif

$ singularity verify --sif-id 1 my_container.sif
Verifying image: my_container.sif
[LOCAL]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[LOCAL]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
```

(continues on next page)

(continued from previous page)

```
1   |1       |NONE    |Def.FILE
Container verified: my_container.sif
```

Note that running the `verify` command without specifying the specific sif-id gives a fatal error. The container is not considered verified as whole because other objects could have been changed without my knowledge.

```
$ singularity verify my_container.sif
Verifying image: my_container.sif
[LOCAL]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[LOCAL]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448

Error encountered during signature verification: object 2: object not signed
FATAL:   Failed to verify container: integrity: object 2: object not signed
```

I can sign a group of objects with the `--group-id` option to `sign`.

```
$ singularity sign --groupid 1 my_container.sif
Signing image: my_container.sif
Enter key passphrase :
Signature created and applied to my_container.sif
```

This creates one signature over all objects in the group. I can verify that nothing in the group has been modified by running `verify` with the same `--group-id` option.

```
$ singularity verify --group-id 1 my_container.sif
Verifying image: my_container.sif
[LOCAL]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[LOCAL]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
1   |1       |NONE    |Def.FILE
2   |1       |NONE    |JSON.Generic
3   |1       |NONE    |FS
Container verified: my_container.sif
```

Because every object in the SIF file is within the signed group 1 the entire container is signed, and the default `verify` behavior without specifying `--group-id` can also verify the container:

```
$ singularity verify my_container.sif
Verifying image: my_container.sif
[LOCAL]   Signing entity: Joe User (Demo keys) <myuser@example.com>
[LOCAL]   Fingerprint: 65833F473098C6215E750B3BDFD69E5CEE85D448
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
1   |1       |NONE    |Def.FILE
2   |1       |NONE    |JSON.Generic
3   |1       |NONE    |FS
Container verified: my_container.sif
```

3.1.3 PEM Key / X.509 Certificate Support

Beginning with version 3.11, SingularityCE supports signing SIF container images using a PEM format private key, and verifying with a PEM format public key, or X.509 certificate. Non-PGP signatures are implemented using the Dead Simple Signing Envelope (DSSE) standard.

Signing with a PEM key

To sign a container using a private key in PEM format, provide the key material to the `sign` command using the `--key` flag:

```
$ singularity sign --key mykey.pem lolcow.sif
INFO:    Signing image with key material from 'mykey.pem'
INFO:    Signature created and applied to image 'lolcow.sif'
```

The DSSE signature descriptor can now be seen by inspecting the SIF file:

```
$ singularity sif list lolcow.sif
--------------------------------------------------------------------------------
ID   |GROUP   |LINK    |SIF POSITION (start-end)  |TYPE
--------------------------------------------------------------------------------
1    |1       |NONE    |32176-32393               |Def.FILE
2    |1       |NONE    |32393-33522               |JSON.Generic
3    |1       |NONE    |33522-33718               |JSON.Generic
4    |1       |NONE    |36864-84656128            |FS (Squashfs/*System/amd64)
5    |NONE    |1   (G) |84656128-84658191         |Signature (SHA-256)

$ singularity sif dump 5 lolcow.sif | jq
{
"payloadType": "application/vnd.sylabs.sif-metadata+json",
...
```

Attempting to `verify` the image without options will fail, as it is not signed with a PGP key:

```
$ singularity verify lolcow.sif
INFO:    Verifying image with PGP key material
FATAL:   Failed to verify container: integrity: key material not provided for DSSE␣
↪envelope signature
```

Note that the error message shows that the container image has a DSSE signature present.

Verifying with a PEM key

To verify a container using a PEM public key directly, provide the key material to the `verify` command using the `key` flag:

```
$ singularity verify --key mypublic.pem lolcow.sif
INFO:    Verifying image with key material from 'mypublic.pem'
Objects verified:
ID  |GROUP   |LINK    |TYPE
------------------------------------------------
1   |1       |NONE    |Def.FILE
```

(continues on next page)

(continued from previous page)

```
2   |1       |NONE    |JSON.Generic
3   |1       |NONE    |JSON.Generic
4   |1       |NONE    |FS
INFO:    Verified signature(s) from image 'lolcow.sif'
```

Verifying with an X.509 certificate

To verify a container that was signed with a PEM private key, using an X.509 certificate, pass the certificate to the `verify` command using the `--certificate` flag. If the certificate is part of a chain, provide intermediate and valid root certificates with the `--certificate-intermediates` and `--certificate-roots` flags:

```
$ singularity verify \
   --certificate leaf.pem \
   --certificate-intermediates intermediate.pem \
   --certificate-roots root.pem \
   lolcow.sif
```

Note: The certificate must have a usage field that allows code signing in order to verify container images.

OSCP Certificate Revocation Checks

When verifying a container using X.509 certificates, SingularityCE can perform online revocation checks using the Online Certificate Status Protocol (OCSP). To enable OCSP checks, add the `--ocsp-verify` flag to your `verify` command:

```
$ singularity verify \
   --certificate leaf.pem \
   --certificate-intermediates intermediate.pem \
   --certificate-roots root.pem \
   --ocsp-verify
   lolcow.sif
```

SingularityCE will then attempt to contact the prescribed OCSP responder for each certificate in the chain, in order to check that the relevant certificate has not been revoked. In the event that an OCSP responder cannot be contacted, or a certificate has been revoked, verification will fail with a validation error:

```
INFO:    Validate: cert:leaf  issuer:intermediate
FATAL:   Failed to verify container: OCSP verification has failed
```

3.2 Key commands

SingularityCE 3.2 introduces the abilities to import, export and remove PGP keys following the OpenPGP standard via GnuPGP (GPG). These commands only modify the local keyring and are not related to the cloud keystore.

3.2.1 Changes in SingularityCE 3.7

SingularityCE 3.7 introduces a global keyring which can be managed by administrators with the new `--global` option. This global keyring is used by ECL (https://sylabs.io/guides/4.0/admin-guide/configfiles.html#ecl-toml) and allows administrators to manage public keys used during ECL image verification.

3.2.2 Key import command

SingularityCE 3.2 allows you import keys reading either from binary or armored key format and automatically detect if it is a private or public key and add it to the correspondent local keystore.

To give a quick view on how it works, we will first consider the case in which a user wants to import a secret (private) key to the local keystore.

First we will check what's the status of the local keystore (which keys are stored by the moment before importing a new key).

```
$ singularity key list --secret
```

Note: Remember that using `--secret` flag will return the secret or private local keyring as output.

Note: The `--private` flag is available as a synonym for `--secret`, as is the shorthand `-s`.

The output will look as it follows:

```
Private key listing (/home/joana/.singularity/sypgp/pgp-secret):

0)  User:               Johnny Cash (none) <cash@example.com>
    Creation time:      2019-04-11 22:22:28 +0200 CEST
    Fingerprint:        47282BDC661F58FA4BEBEF47CA576CBD8EF1A2B4
    Length (in bits):   3072

1)  User:               John Green (none) <john@example.com>
    Creation time:      2019-04-11 13:08:45 +0200 CEST
    Fingerprint:        5720799FE7B048CF36FAB8445EE1E2BD7B6342C5
    Length (in bits):   1024
```

Note: Remember that running that same command but with sudo privilege, will give you a totally different list since it will be the correspondent keystore from user `root`

After this, you can simply import the key you need by adding the exact location to the file, let's say you own a gpg key file named `pinkie-pie.asc` which is a secret GPG key you want to import. Then you will just need to run the following command to import your key:

```
$ singularity key import $HOME/pinkie-pie.asc
```

Note: This location is considering your key was located on the `$HOME` directory. You can specify any location to the file.

Since you're importing a private (secret) key, you will need to specify the passphrase related to it and then a new passphrase to be added on your local keystore.

```
Enter your old password :
Enter a new password for this key :
Retype your passphrase :
Key with fingerprint 8C10B902F438E4D504C3ACF689FCFFAED5F34A77 successfully added to the␣
↪keyring
```

After this you can see if that key was correctly added to your local keystore by running `singularity key list -s` command:

```
Private key listing (/home/joana/.singularity/sypgp/pgp-secret):

1)  User:               Johnny Cash (none) <cash@example.com>
    Creation time:      2019-04-11 22:22:28 +0200 CEST
    Fingerprint:        47282BDC661F58FA4BEBEF47CA576CBD8EF1A2B4
    Length (in bits):   3072

2)  User:               John Green (none) <john@example.com>
    Creation time:      2019-04-11 13:08:45 +0200 CEST
    Fingerprint:        5720799FE7B048CF36FAB8445EE1E2BD7B6342C5
    Length (in bits):   1024

3)  User:               Pinkie Pie (Eternal chaos comes with chocolate rain!)
↪<balloons@example.com>
    Creation time:      2019-04-26 12:07:07 +0200 CEST
    Fingerprint:        8C10B902F438E4D504C3ACF689FCFFAED5F34A77
    Length (in bits):   1024
```

You will see the imported key at the bottom of the list. Remember you can also import an `ascii` armored key and this will be automatically detected by the `key import` command (no need to specify the format).

Note: In case you would like to import a public key the process remains the same, as the import command will automatically detect whether this key to be imported is either public or private.

3.2.3 Key export command

The key export command allows you to export a key that is on your local keystore. This key could be either private or public, and the key can be exported on `ASCII` armored format or on binary format. Of course to identify the keystore and the format the syntax varies from the `key import` command.

For example to export a public key in binary format you can run:

```
$ singularity key export 8C10B902F438E4D504C3ACF689FCFFAED5F34A77 $HOME/mykey.asc
```

This will export a public binary key named `mykey.asc` and will save it under the home folder. If you would like to export the same public key but in an `ASCII` armored format, you would need to run the following command:

```
$ singularity key export --armor 8C10B902F438E4D504C3ACF689FCFFAED5F34A77 $HOME/mykey.asc
```

And in the case in which you may need to export a secret key on `ASCII` armored format, you would need to specify from where to find the key, since the fingerprint is the same.

```
$ singularity key export --armor --secret 8C10B902F438E4D504C3ACF689FCFFAED5F34A77 $HOME/
↪mykey.asc
```

and on binary format instead:

```
$ singularity key export --secret 8C10B902F438E4D504C3ACF689FCFFAED5F34A77 $HOME/mykey.
↪asc
```

Note: Exporting keys will not change the status of your local keystore or keyring. This will just obtain the content of the keys and save it on a local file on your host.

Note: The `--private` flag is available as a synonym for `--secret`, as is the shorthand `-s`.

3.2.4 Key remove command

In case you would want to remove a public key from your public local keystore, you can do so by running the following command:

```
$ singularity key remove 8C10B902F438E4D504C3ACF689FCFFAED5F34A77
```

Note: Remember that this will only delete the public key and not the private one with the same matching fingerprint.

If you want to remove a private key from your local keystore, you can do so by passing the *–secret* flag to *key remove*:

```
$ singularity key remove --secret 8C10B902F438E4D504C3ACF689FCFFAED5F34A77
```

Note: Remember that this will only delete the private key and not the public one with the same matching fingerprint.

Note: The `--private` flag is available as a synonym for `--secret`, as is the shorthand `-s`.

3.3 Encrypted Containers

Users can build a secure, confidential container environment by encrypting the root file system.

3.3.1 Overview

In SingularityCE >= v3.4.0 a new feature to build and run encrypted containers has been added to allow users to encrypt the file system image within a SIF. This encryption can be performed using either a passphrase or asymmetrically via an RSA key pair in Privacy Enhanced Mail (PEM/PKCS1) format. The container is encrypted in transit, at rest, and even while running. In other words, there is no intermediate, decrypted version of the container on disk. Container decryption occurs at runtime completely within kernel space.

Note: This feature utilizes the Linux `dm-crypt` library and `cryptsetup` utility and requires cryptsetup version of >= 2.0.0. This version should be standard with recent Linux versions such as, but users of older Linux versions may have to update.

3.3.2 Encrypting a container

A container can be encrypted either by supplying a plaintext passphrase or a PEM file containing an asymmetric RSA public key. Of these two methods the PEM file is more secure and is therefore recommended for production use.

Note: In SingularityCE 3.4, the definition file stored with the container will not be encrypted. If it contains sensitive information you should remove it before encryption via `singularity sif del 1 myimage.sif`. Metadata encryption will be addressed in a future release.

An `-e|--encrypt` flag to `singularity build` is used to indicate that the container needs to be encrypted.

A passphrase or a key-file used to perform the encryption is supplied at build time via an environment variable or a command line option.

Encryption Method	Environment Variable	Commandline Option
Passphrase	`SINGULARITY_ENCRYPTION_PASSPHRASE`	`--passphrase`
Asymmetric Key (PEM)	`SINGULARITY_ENCRYPTION_PEM_PATH`	`--pem-path`

The `-e|--encrypt` flag is implicitly set when the `--passphrase` or `--pem-path` flags are passed with the build command. If multiple encryption related flags and/or environment variables are set, the following precedence is respected.

1. `--pem-path`
2. `--passphrase`
3. `SINGULARITY_ENCRYPTION_PEM_PATH`
4. `SINGULARITY_ENCRYPTION_PASSPHRASE`

Passphrase Encryption

Note: Passphrase encryption is less secure than encrypting containers using an RSA key pair (detailed below). Passphrase encryption is provided as a convenience, and as a way for users to familiarize themselves with the encrypted container workflow, but users running encrypted containers in production are encouraged to use asymmetric keys.

In case of plaintext passphrase encryption, a passphrase is supplied by one of the following methods.

Encrypting with a passphrase interactively

```
$ sudo singularity build --passphrase encrypted.sif encrypted.def
Enter encryption passphrase: <secret>
INFO:    Starting build...
```

Using an environment variable

```
$ sudo SINGULARITY_ENCRYPTION_PASSPHRASE=<secret> singularity build --encrypt encrypted.
↪sif encrypted.def
Starting build...
```

In this case it is necessary to use the `--encrypt` flag since the presence of an environment variable alone will not trigger the encrypted build workflow.

While this example shows how an environment variable can be used to set a passphrase, you should set the environment variable in a way that will not record your passphrase on the command line. For instance, you could save a plain text passphrase in a file (e.g. `secret.txt`) and use it like so.

```
$ export SINGULARITY_ENCRYPTION_PASSPHRASE=$(cat secret.txt)

$ sudo -E singularity build --encrypt encrypted.sif encrypted.def
Starting build...
```

PEM File Encryption

SingularityCE currently supports RSA encryption using a public/private key-pair. Keys are supplied in PEM format. The public key is used to encrypt containers that can be decrypted on a host that has access to the secret private key.

You can create a pair of RSA keys suitable for encrypting your container with the `ssh-keygen` command, and then create a PEM file with a few specific flags like so:

```
# Generate a key pair
$ ssh-keygen -t rsa -b 2048 -m pem -N ''
Generating public/private rsa key pair.
Enter file in which to save the key (/home/vagrant/.ssh/id_rsa): rsa
Your identification has been saved in rsa
Your public key has been saved in rsa.pub
The key fingerprint is:
[snip...]
```

(continues on next page)

(continued from previous page)

```
# Convert the public key to PEM PKCS1 format
$ ssh-keygen -f ./rsa.pub -e -m pem >rsa_pub.pem

# Rename the private key (already PEM PKCS1) to a nice name
$ mv rsa rsa_pri.pem
```

You would use the `rsa_pub.pem` file to encrypt your container and the `rsa_pri.pem` file to run it.

Encrypting with a command line option

```
$ sudo singularity build --pem-path=rsa_pub.pem encrypted.sif encrypted.def
Starting build...
```

Encrypting with an environment variable

```
$ sudo SINGULARITY_ENCRYPTION_PEM_PATH=rsa_pub.pem singularity build --encrypt encrypted.
↪sif encrypted.def
Starting build...
```

In this case it is necessary to use the `--encrypt` flag since the presence of an environment variable alone will not trigger the encrypted build workflow.

3.3.3 Running an encrypted container

To `run`, `shell`, or `exec` an encrypted image, credentials to decrypt the image need to be supplied at runtime either in a key-file or a plaintext passphrase.

Running a container encrypted with a passphrase

A passphrase can be supplied at runtime by either of the ways listed in the sections above.

Running with a passphrase interactively

```
$ singularity run --passphrase encrypted.sif
Enter passphrase for encrypted container: <secret>
```

Running with a passphrase in an environment variable

```
$ SINGULARITY_ENCRYPTION_PASSPHRASE="secret" singularity run encrypted.sif
```

While this example shows how an environment variable can be used to set a passphrase, you should set the environment variable in a way that will not record your passphrase on the command line. For instance, you could save a plain text passphrase in a file (e.g. `secret.txt`) and use it like so.

```
$ export SINGULARITY_ENCRYPTION_PASSPHRASE=$(cat secret.txt)

$ singularity run encrypted.sif
```

Running a container encrypted with a PEM file

A private key is supplied using either of the methods listed in the Encryption section above.

Running using a command line option

```
$ singularity run --pem-path=rsa_pri.pem encrypted.sif
```

Running using an environment variable

```
$ SINGULARITY_ENCRYPTION_PEM_PATH=rsa_pri.pem singularity run encrypted.sif
```

CHAPTER

FOUR

SHARING & ONLINE SERVICES

Sylabs offers a suite of container services, with a free tier and on-premise options. Learn how to make use these services to simplify the process of building, signing, and sharing your containers.

4.1 Remote Endpoints

4.1.1 Overview

The `remote` command group allows users to manage the remote endpoint(s) that SingularityCE interacts with (Singularity Container Services, or a local Singularity Enterprise installation).

Note: In previous versions of SingularityCE, the `remote` command group also included subcommands for interacting with OCI registries (for image storage services), as well as keyservers (used to locate public keys for SIF image verification). Beginning with version 4.0, this functionality has been moved to dedicated top-level command groups: *registry* and *keyserver*, respectively.

4.1.2 Singularity Container Services

Sylabs introduced Singularity Container Services (SCS) to enable users to Create container images, Secure them, and Share them with others.

A fresh, default installation of SingularityCE is configured to connect to the public services available at cloud.sylabs.io. If you only want to use these public services, all you need to do is obtain an authentication token, which you then provide to `singularity remote login`:

1. Go to: https://cloud.sylabs.io/
2. Click "Sign In" and follow the sign-in steps.
3. Click on your login id, which, after sign-in, should appear on the right side of the navigation-bar at the top of the page.
4. Select "Access Tokens" from the drop down menu.
5. Enter a name for your new access token, such as "test token".
6. Click the "Create a New Access Token" button.
7. Click "Copy token to Clipboard" from the "New API Token" page.
8. Run `singularity remote login` and paste the access token when prompted.

Once your token is stored, you can check that you are able to connect to the services by using the `status` subcommand:

```
$ singularity remote status
INFO:    Checking status of default remote.
SERVICE    STATUS  VERSION                  URI
Builder    OK      v1.6.9-rc.4-0-g87336319  https://build.sylabs.io
Consent    OK      v1.7.0-0-g66ba1a9        https://auth.sylabs.io/consent
Keyserver  OK      v1.18.12-0-gab541fb      https://keys.sylabs.io
Library    OK      v0.3.8-rc.6-0-g630cdaa   https://library.sylabs.io
Token      OK      v1.7.0-0-g66ba1a9        https://auth.sylabs.io/token

Logged in as: myname <myemail@example.com>

INFO:    Access Token Verified!

Valid authentication token set (logged in).
```

If you see any errors, you may need to check if your system requires the setting of environment variables for a network proxy, or if a firewall may be blocking access to `*.sylabs.io`. Consult your system administrator.

You can interact with Singularity Container Services using various SingularityCE commands:

pull, push, build –remote, key, search, verify, exec, shell, run, instance

---

Note: Using the commands listed above will *not* interact with SCS if given URIs beginning with `docker://`, `oras://` or `shub://`.

4.1.3 Managing Remote Endpoints

Users can set up and switch between multiple remote endpoints, which will be stored in their `~/.singularity/remote.yaml` file. Alternatively, remote endpoints can be set on a system-wide basis by an administrator.

A remote endpoint may be Singularity Container Services, a private installation of Singularity Enterprise, or any community-developed service that is API-compatible.

Generally, users and administrators should manage remote endpoints using the `singularity remote` command, and avoid editing `remote.yaml` configuration files directly.

Listing and Logging In to Remote Endpoints

To `list` existing remote endpoints, run the following:

```
$ singularity remote list

NAME         URI              DEFAULT?  GLOBAL?  EXCLUSIVE?  SECURE?
SylabsCloud  cloud.sylabs.io  ✓         ✓                    ✓
```

The ✓ in the `DEFAULT?` column for `SylabsCloud` shows that this is the current default remote endpoint.

To `login` to a remote for the first time, or when a token needs to be replaced (if it has expired or been revoked), run the following:

```
# Login to the default remote endpoint
$ singularity remote login
```

(continues on next page)

(continued from previous page)

```
# Login to another remote endpoint
$ singularity remote login <remote_name>

# example:
$ singularity remote login SylabsCloud
Generate an access token at https://cloud.sylabs.io/auth/tokens, and paste it here.
Token entered will be hidden for security.
Access Token:
INFO:    Access Token Verified!
INFO:    Token stored in /home/myuser/.singularity/remote.yaml
```

If you `login` to a remote that you already have a valid token for, you will be prompted for confirmation that you indeed want to replace the current token, and the new token will be verified before it replaces your existing credential. If you enter an incorrect token your existing token will not be replaced,

```
$ singularity remote login
An access token is already set for this remote. Replace it? [y/N] y
Generate an access token at https://cloud.sylabs.io/auth/tokens, and paste it here.
Token entered will be hidden for security.
Access Token:
FATAL:   while verifying token: error response from server: Invalid Credentials

# Previous token is still in place
```

Note: It is important for users to be aware that the `remote login` command will store the supplied credentials or tokens **unencrypted** in your home directory. Please ensure that the access permissions on your home directory are set accordingly, to protect your credentials from unwanted access.

Adding and Removing Remote Endpoints

To `add` a remote endpoint (for the current user only):

```
$ singularity remote add <remote_name> <remote_uri>
```

For example, if you have an installation of SingularityCE enterprise hosted at enterprise.example.com:

```
$ singularity remote add myremote https://enterprise.example.com
INFO:    Remote "myremote" added.
Generate an access token at https://enterprise.example.com/auth/tokens, and paste it␣
↪here.
Token entered will be hidden for security.
Access Token:
```

You will be prompted to setup an API key as the remote is added. As the example above shows, the output of the `add` subcommand will provide you with the web address you need to visit in order to generate your new access token.

To `add` a global remote endpoint (available to all users on the system), an administrative user should run:

```
$ sudo singularity remote add --global <remote_name> <remote_uri>
```

(continues on next page)

(continued from previous page)

```
# example...
$ sudo singularity remote add --global company-remote https://enterprise7.example.com
INFO:    Remote "company-remote" added.
INFO:    Global option detected. Will not automatically log into remote.
```

Note: Global remote configurations can only be modified by the root user, and are stored in the `etc/singularity/remote.yaml` file under the SingularityCE installation directory.

Conversely, to `remove` an endpoint:

```
$ singularity remote remove <remote_name>
```

Use the `--global` option as the root user to remove a global endpoint:

```
$ sudo singularity remote remove --global <remote_name>
```

Insecure (HTTP) Endpoints

Starting with SingularityCE 3.9, if you are using a endpoint that only exposes its service discovery file over an insecure HTTP connection, it can be added by specifying the `--insecure` flag:

```
$ sudo singularity remote add --global --insecure test http://test.example.com
INFO:    Remote "test" added.
INFO:    Global option detected. Will not automatically log into remote.
```

This flag causes HTTP to be used instead of HTTPS *for service discovery only*. The protocol used to access individual library-, build- and keyservice-URLs is determined by the contents of the service discovery file.

Set the Default Remote

To use a given remote endpoint as the default for commands such as `push`, `pull`, etc., use the `remote use` command:

```
$ singularity remote use <remote_name>
```

The remote designated as default shows up with a `YES` under the `ACTIVE` column in the output of `remote list`:

```
$ singularity remote list

NAME            URI                       DEFAULT?  GLOBAL?  EXCLUSIVE?  SECURE?
SylabsCloud     cloud.sylabs.io                     ✓                    ✓
company-remote  enterprise7.example.com             ✓                    ✓
myremote        enterprise.example.com    ✓                              ✓
test            test.example.com                    ✓                    ✓

$ singularity remote use SylabsCloud
INFO:    Remote "SylabsCloud" now in use.

$ singularity remote list
```

(continues on next page)

(continued from previous page)

```
NAME            URI                       DEFAULT?  GLOBAL?  EXCLUSIVE?  SECURE?
SylabsCloud     cloud.sylabs.io           ✓         ✓                    ✓
company-remote  enterprise7.example.com             ✓                    ✓
myremote        enterprise.example.com                                   ✓
test            test.example.com                    ✓                    ✓
```

In the example above, the default remote at the start (before being changed to `SylabsCloud`) was `myremote`. That is because adding a new remote endpoint automatically makes the newly-added endpoint the default one, and the same user had previously used the `remote add` command to add the `myremote` endpoint. This behavior can be suppressed by passing the `--no-default` flag to the `remote add` command, which will then add a new remote endpoint but leave the default endpoint unchanged:

```
$ singularity remote add --no-default myotherremote https://enterprise2.example.com
INFO:    Remote "myotherremote" added.
Generate an access token at https://enterprise2.example.com/auth/tokens, and paste it␣
↪here.
Token entered will be hidden for security.
Access Token:

$ singularity remote list

NAME            URI                       DEFAULT?  GLOBAL?  EXCLUSIVE?  SECURE?
SylabsCloud     cloud.sylabs.io           ✓         ✓                    ✓
company-remote  enterprise7.example.com             ✓                    ✓
myotherremote   enterprise2.example.com                                   ✓
myremote        enterprise.example.com                                   ✓
test            test.example.com                    ✓                    ✓
```

SingularityCE 3.7 introduces the ability for an administrator to make a remote the only usable remote for the system, using the `--exclusive` flag:

```
$ sudo singularity remote use --exclusive company-remote
INFO:    Remote "company-remote" now in use.

$ singularity remote list

NAME            URI                       DEFAULT?  GLOBAL?  EXCLUSIVE?  SECURE?
SylabsCloud     cloud.sylabs.io                     ✓                    ✓
company-remote  enterprise7.example.com   ✓         ✓        ✓           ✓
myotherremote   enterprise2.example.com                                   ✓
myremote        enterprise.example.com                                   ✓
test            test.example.com                    ✓                    ✓
```

This, in turn, prevents users from changing the remote they use:

```
$ singularity remote use myremote
FATAL:   could not use myremote: remote company-remote has been set exclusive by the␣
↪system administrator
```

If you do not want to switch remote with `remote use`, you can:

- Instruct `push` and `pull` commands to use an alternative library server using the `--library` option (for example: `singularity pull -F --library https://library.example.com library://alpine`). Note

that the URL provided to the `--library` option is the URL of the library service itself, not the service discovery URL for the entire remote.

- Instruct the `build --remote` commands to use an alternative remote builder using the `--builder` option.
- Instruct certain subcommands of the `key` command to use an alternative keyserver using the `--url` option (for example: `singularity key search --url https://keys.example.com foobar`).

4.2 Keyserver Management

By default, SingularityCE will use the keyserver defined by the active *remote endpoint*'s service discovery file. This behavior can be changed or supplemented using the `key` command and, in particular, its subcommands `add` and `remove`. These allow an administrator to create a global list of keyservers that will be used to verify container signatures by default. When verifying container signatures, SingularityCE consults them according to a configured *order* (with the keyserver whose order is 1 consulted first, then the one whose order is 2, and so forth). Other operations performed by SingularityCE that reach out to a keyserver will only use the first one (whose order is 1).

Note: In previous versions of SingularityCE, the functionality described here was grouped together with *remote endpoint management* under the `remote` command group. Beginning with version 4.0, this functionality has been given its own top-level command group, `keyserver`.

The `list` subcommand allows the user to examine the set of currently configured keyservers:

```
$ singularity keyserver list

SylabsCloud*^
   #1  https://keys.sylabs.io  TLS

(* = system endpoint, ^ = default endpoint,
 + = user is logged in directly to this keyserver)
```

We can see in the output of the `list` subcommand that "SylabsCloud" is the *default* remote endpoint (in other words, the endpoint that will be used by all SingularityCE commands unless otherwise specified), and that it is a *global* (in other words, system-level) endpoint. As can be seen above, the output also indicates that TLS will be used when communicating with the `https://keys.sylabs.io` keyserver.

We can add a key server to list of keyservers as follows:

```
$ sudo singularity keyserver add https://pgp.example.com
$ singularity keyserver list

SylabsCloud*^
   #1  https://keys.sylabs.io   TLS
   #2  https://pgp.example.com  TLS

(* = system endpoint, ^ = default endpoint,
 + = user is logged in directly to this keyserver)
```

Here, we see that the `https://pgp.example.com` keyserver was added to the list. We can specify the order in the list in which this keyserver should be added, by using the `--order` flag:

```
$ sudo singularity keyserver add --order 1 https://pgp.example.com
$ singularity keyserver list
```

(continues on next page)

(continued from previous page)

```
SylabsCloud*^
   #1  https://pgp.example.com  TLS
   #2  https://keys.sylabs.io   TLS

(* = system endpoint, ^ = default endpoint,
 + = user is logged in directly to this keyserver)
```

Since we specified `--order 1`, the `https://pgp.example.com` keyserver was added as the first entry in the list, and the default keyserver was moved to second in the list. With this keyserver configuration, all default image verification performed by SingularityCE will, when searching for public keys, reach out to `https://pgp.example.com` first, and only then to `https://keys.sylabs.io`.

If a keyserver requires authentication prior to being used, users can login as follows, supplying the password or an API token at the prompt:

```
$ singularity keyserver login --username myname https://pgp.example.com
Password / Token:
INFO:    Token stored in /home/myuser/.singularity/remote.yaml
```

The output of *keyserver list* will now show that we are logged in to `https://pgp.example.com`:

```
$ singularity keyserver list

SylabsCloud *^
   #1  https://pgp.example.com          TLS  +
   #2  https://keys.sylabs.io           TLS

(* = system endpoint, ^ = default endpoint,
 + = user is logged in directly to this keyserver)
```

Note: It is important for users to be aware that the `keyserver login` command will store the supplied credentials or tokens unencrypted in your home directory.

4.3 OCI Image Registries

It is common for users of SingularityCE to use OCI registries as sources for their container images. Some registries require credentials to access certain images or even the registry itself. Previously, the only method in SingularityCE to supply credentials to registries was to supply credentials for each command or set environment variables to contain the credentials for a single registry. See Authentication via Interactive Login and Authentication via Environment Variables.

Starting with SingularityCE 4.0, users can supply credentials on a per-registry basis with the `registry` command.

Note: In versions of SingularityCE starting from 3.7 but before 4.0, the functionality described here was grouped together with *remote endpoint management* under the `remote` command group. Beginning with version 4.0, this functionality has been given its own top-level command group, `registry`.

Users can login to an OCI registry with the `registry login` command by specifying a `docker://` prefix to the registry hostname:

```
$ singularity registry login --username myuser docker://docker.com
Password / Token:
INFO:    Token stored in /home/myuser/.singularity/remote.yaml

$ singularity registry list

URI                  SECURE?
docker://docker.com  ✓
```

SingularityCE will automatically supply the configured credentials when interacting with DockerHub. The checkmark in the SECURE? column indicates that SingularityCE will use TLS when communicating with the registry.

A user can be logged-in to multiple OCI registries at the same time:

```
$ singularity registry login --username myuser docker://registry.example.com
Password / Token:
INFO:    Token stored in /home/myuser/.singularity/remote.yaml

$ singularity registry list

URI                            SECURE?
docker://docker.com            ✓
docker://registry.example.com  ✓
```

SingularityCE will supply the correct credentials for the registry based on the hostname used, whenever one of the following commands is used with a `docker://` or `oras://` URI:

pull, push, build, exec, shell, run, instance

Note: It is important for users to be aware that the `registry login` command will store the supplied credentials or tokens unencrypted in your home directory.

4.4 SCS Library

4.4.1 Overview

The Singularity Container Services (SCS) Library is the place to *push* your containers to the cloud so other users can *pull*, *verify*, and use them.

SCS also provides a *Remote Builder*, allowing you to build containers on a secure remote service. This is convenient so that you can build containers on systems where you do not have root privileges.

4.4.2 Make an Account

Making an account is easy, and straightforward:

1. Go to: https://cloud.sylabs.io/library.
2. Click "Sign in to Sylabs" (top right corner).
3. Select your method to sign in, with Google, GitHub, GitLab, or Microsoft.
4. Type your passwords, and that's it!

4.4.3 Creating a Access token

Access tokens for pushing a container, and remote builder.

To generate a access token, do the following steps:

1. Go to: https://cloud.sylabs.io/
2. Click "Sign In" and follow the sign in steps.
3. Click on your login id (same and updated button as the Sign in one).
4. Select "Access Tokens" from the drop down menu.
5. Enter a name for your new access token, such as "test token"
6. Click the "Create a New Access Token" button.
7. Click "Copy token to Clipboard" from the "New API Token" page.
8. Run `singularity remote login` and paste the access token at the prompt.

Now that you have your token, you are ready to push your container!

4.4.4 Pushing a Container

The `singularity push` command will push a container to the container library with the given URL. Here's an example of a typical push command:

```
$ singularity push my-container.sif library://your-name/project-dir/my-container:latest
```

The `:latest` is the container tag. Tags are used to have different version of the same container.

Note: When pushing your container, there's no need to add a `.sif` (Singularity Image Format) to the end of the container name, (like on your local machine), because all containers on the library are SIF containers.

Let's assume you have your container (v1.0.1), and you want to push that container without deleting your `:latest` container, then you can add a version tag to that container, like so:

```
$ singularity push my-container.sif library://your-name/project-dir/my-container:1.0.1
```

You can download the container with that tag by replacing the `:latest`, with the tagged container you want to download.

To set a description against the container image as you push it, use the *-D* flag introduced in SingularityCE 3.7. This provides an alternative to setting the description via the web interface:

```
$ singularity push -D "My alpine 3.11 container" alpine_3.11.sif library://myuser/
↪examples/alpine:3.11
2.7MiB / 2.7MiB␣
↪[=============================================================================] 100 % 1.1␣
↪MiB/s 0s

Library storage: using 13.24 MiB out of 11.00 GiB quota (0.1% used)
Container URL: https://cloud.sylabs.io/library/myuser/examples/alpine
```

Note that when you push to a library that supports it, SingularityCE 3.7 and above will report your quota usage and the direct URL to view the container in your web browser.

OCI-SIF Images

If you are using SingularityCE 4's new OCI-mode, you may wish to push OCI-SIF images to a `library://` destination. The standard `push` command can be used, and SingularityCE will perform the push as an OCI image.

Instead of uploading the container as a single SIF file, the OCI configuration and layer blobs that are encapsulated in the OCI-SIF will be uploaded to the OCI registry that sits behind the SCS / Singularity Enterprise Library.

Note: The OCI image specification doesn't support SIF signatures, or any additional partitions that can be added to SIF (including OCI-SIF) files by SingularityCE.

If you have signed an OCI-SIF image locally, the signature(s) will not be pushed to the library. You may wish to push the OCI-SIF, as a single file, to an OCI registry using the `oras://` protocol instead.

Pushing OCI-SIF containers to the library in this manner means that they can be accessed by other OCI tools. For example, you can use the Skopeo CLI tool to examine the image in the registry after it has been pushed. First, `push` an OCI-SIF to the SCS `library://`. The `-U` option is needed because the image is unsigned.

```
$ singularity push -U alpine_latest.oci.sif library://example/userdoc/alpine:latest
WARNING: Skipping container verification
INFO:    Pushing an OCI-SIF to the library OCI registry. Use `--oci` to pull this image.
```

Now use `skopeo` to access the image in the library. This requires authentication, which is handled automatically when you use `singularity push`. For other tools, SingularityCE provides a command `singularity remote get-login-password` that will provide a token that we can use to login to `registry.sylabs.io`, which is the address of the OCI registry backing the SCS library.

```
$ singularity remote get-login-password | \
    skopeo login -u example --password-stdin registry.sylabs.io
Login Succeeded!
```

Finally, use `skopeo inspect` to examine the image pushed earlier:

```
$ skopeo inspect docker://registry.sylabs.io/example/userdoc/alpine:latest
{
   "Name": "registry.sylabs.io/example/userdoc/alpine",
   "Digest":
↪"sha256:d08ad9745675812310727c0a99a4472b82fb1cc81e5c42ceda023f1bc35ca34a",
   "RepoTags": [
      "latest"
```

(continues on next page)

(continued from previous page)

```
   ],
   "Created": "2023-08-07T20:16:26.309461618Z",
   "DockerVersion": "",
   "Labels": null,
   "Architecture": "amd64",
   "Os": "linux",
   "Layers": [
      "sha256:a0c5ced3a57bd1d0d71aaf4a0ea6131d5f163a4a8c5355468c18d4ef006f5d7d"
   ],
   "LayersData": [
      {
            "MIMEType": "application/vnd.sylabs.image.layer.v1.squashfs",
            "Digest":
↪"sha256:a0c5ced3a57bd1d0d71aaf4a0ea6131d5f163a4a8c5355468c18d4ef006f5d7d",
            "Size": 3248128,
            "Annotations": null
      }
   ],
   "Env": [
      "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
   ]
}
```

Because the OCI-SIF was pushed as an OCI image, `skopeo inspect` is able to show the image configuration. This is not possible for non-OCI-SIF images:

```
$ skopeo inspect docker://registry.sylabs.io/library/default/alpine:latest
FATA[0001] unsupported image-specific operation on artifact with type "application/vnd.
↪sylabs.sif.config.v1+json"
```

4.4.5 Pulling a container

The `singularity pull` command will download a container from the Library (`library://`), Docker Hub (`docker://`), and also Shub (`shub://`).

Note: When pulling from Docker, the container will automatically be converted to a SIF (Singularity Image Format) container.

Here's a typical pull command:

```
$ singularity pull file-out.sif library://alpine:latest

# or pull from docker:

$ singularity pull file-out.sif docker://alpine:latest
```

Note: If there's no tag after the container name, SingularityCE automatically will pull the container with the `:latest` tag.

To pull a container with a specific tag, just add the tag to the library URL:

```
$ singularity pull file-out.sif library://alpine:3.8
```

Of course, you can pull your own containers. Here's what that will look like:

Pulling your own container

Pulling your own container is just like pulling from Github, Docker, etc…

```
$ singularity pull out-file.sif library://your-name/project-dir/my-container:latest

# or use a different tag:

$ singularity pull out-file.sif library://your-name/project-dir/my-container:1.0.1
```

Note: You *don't* have to specify a output file, one will be created automatically, but it's good practice to always specify your output file.

OCI-SIF Images

If you are using SingularityCE 4's new OCI-mode and have pushed OCI-SIF containers to the SCS library, they are stored as OCI images in the OCI registry that backs the library. You can pull these images with the standard `pull` command:

```
$ singularity pull library://sylabs/test/alpine-oci-sif:latest
INFO:    sylabs/test/alpine-oci-sif:latest is an OCI image, attempting to fetch as an␣
↪OCI-SIF
Getting image source signatures
Copying blob af32528d4445 done
Copying config a5d222bd0d done
Writing manifest to image destination
INFO:    Writing OCI-SIF image
INFO:    Cleaning up.
WARNING: integrity: signature not found for object group 1
WARNING: Skipping container verification
```

Note that SingularityCE detects the image is an OCI image, and automatically retrieves it to an OCI-SIF file.

If the image was a non-OCI-SIF, built for SingularityCE's default native mode, then it would be retrieved as-is. To ensure that an image retrieved from a `library://` URI is an OCI-SIF, use the `--oci` flag. This will produce an error if a non-OCI-SIF is pulled:

```
$ singularity pull --oci library://sylabs/examples/ruby
Getting image source signatures
Copying blob a21814eefb7f done
Copying config 5211e7986c done
Writing manifest to image destination
INFO:    Cleaning up.
FATAL:   While pulling library image: error fetching image: while creating OCI-SIF:␣
↪while checking OCI image: json: cannot unmarshal string into Go struct field␣
↪ConfigFile.rootfs of type v1.RootFS
```

Specifying a platform / architecture

By default, `singularity pull` from a `library://` URI will attempt to fetch a container that matches the architecture of your host system. If you need to retrieve a container that does not have the same architecture as your host (e.g. an `arm64` container on an `amd64` host), you can use the `--platform` or `--arch` options.

The `--arch` option accepts a CPU architecture only. For example, to pull an Ubuntu image for a 64-bit ARM system:

```
$ singularity pull --arch arm64 library://ubuntu
```

The `--platform` option accepts an OCI platform string. This has two or three parts, separated by forward slashes (`/`):

- An OS value. Only `linux` is supported by SingularityCE.
- A CPU architecture value, e.g. `arm64`.
- An optional CPU variant, e.g. `v8`.

Note that the library does not support CPU variants. Any CPU variant provided will be ignored.

To pull an Ubuntu image for a 64-bit ARM system from the library, using the `--platform` option:

```
$ singularity pull --platform linux/arm64 library://ubuntu
```

4.4.6 Verify/Sign your Container

Verify containers that you pull from the library, ensuring they are bit-for-bit reproductions of the original image.

Check out *this page* on how to: *verify a container*, *making PGP key, and sign your own containers*.

4.4.7 Searching the Library for Containers

To find interesting or useful containers in the library, you can open https://cloud.sylabs.io/library in your browser and search from there through the web GUI.

Alternatively, from the CLI you can use `singularity search <query>`. This will search the library for container images matching `<query>`.

Using the CLI Search

Here is an example of searching the library for `centos`:

```
singularity search centos
Found 72 container images for amd64 matching "centos":

    library://dcsouthwick/iotools/centos7:latest

    library://dcsouthwick/iotools/centos7:sha256.
↪48e81523aaad3d74e7af8b154ac5e75f2726cc6cab37f718237d8f89d905ff89
            Minimal centos7 image from yum bootstrap

    library://dtrudg/linux/centos:7,centos7,latest

    library://dtrudg/linux/centos:centos6,6
```

(continues on next page)

(continued from previous page)

```
    library://emmeff/centos/centos:8

    library://essen1999/default/centos-tree:latest

    library://gallig/default/centos_benchmark-signed:7.7.1908
            Signed by: 6B44B0BC9CD273CC6A71DA8CED6FA43EF8771A02

    library://gmk/default/centos7-devel:latest
            Signed by: 7853F08767A4596B3C1AD95E48E1080AB16ED1BC
```

Containers can have multiple tags, and these are shown separated by commas after the `:` in the URL. E.g. `library://dtrudg/linux/centos:7,centos7,latest` is a single container image with 3 tags, `7`, `centos7`, and `latest`. You can `singularity pull` the container image using any one of these tags.

Note that the results show `amd64` containers only. By default `search` returns only containers with an architecture matching your current system. To e.g. search for `arm64` containers from an `amd64` machine you can use the `--arch` flag:

```
singularity search --arch arm64 alpine
Found 5 container images for arm64 matching "alpine":

    library://dtrudg-sylabs-2/multiarch/alpine:latest

    library://geoffroy.vallee/alpine/alpine:latest
            Signed by: 9D56FA7CAFB4A37729751B8A21749D0D6447B268

    library://library/default/alpine:3.11.5,latest,3,3.11

    library://library/default/alpine:3.9,3.9.2

    library://sylabs/tests/passphrase_encrypted_alpine:3.11.5
```

You can also limit results to only signed containers with the `--signed` flag:

```
singularity search --signed alpine
Found 45 container images for amd64 matching "alpine":

    library://deep/default/alpine:latest,1.0.1
            Signed by: 8883491F4268F173C6E5DC49EDECE4F3F38D871E

    library://godloved/secure/alpine:20200514.0.0
            Signed base image built directly from mirrors suitable for secure building.␣
↪Make sure to check that the fingerprint is B7761495F83E6BF7686CA5F0C1A7D02200787921
            Signed by: B7761495F83E6BF7686CA5F0C1A7D02200787921

    library://godlovedc/blah/alpine:sha256.
↪63259fd0a2acb88bb652702c08c1460b071df51149ff85dc88db5034532a14a0
            Signed by: 8883491F4268F173C6E5DC49EDECE4F3F38D871E

    library://heffaywrit/base/alpine:latest
            Signed by: D4038BDDE21017435DFE5ADA9F2D10A25D64C1EF

    library://hellseva/class/alpine:latest
```

(continues on next page)

(continued from previous page)

```
            Signed by: 6D60F95E86A593603897164F8E09E44D12A7111C

    library://hpc110/default/alpine-miniconda:cupy
            Signed by: 9FF48D6202271D3C842C53BD0D237BE8BB5B5C76
    ...
```

4.4.8 Remote Builder

The remote builder service can build your container in the cloud removing the requirement for root access.

Here's a typical remote build command:

```
$ singularity build --remote file-out.sif docker://ubuntu:22.04
```

Building from a definition file:

This is our definition file. Let's call it `ubuntu.def`:

```
bootstrap: library
from: ubuntu:22.04

%runscript
    echo "hello world from ubuntu container!"
```

Now, to build the container, use the `--remote` flag, and without `sudo`:

```
$ singularity build --remote ubuntu.sif ubuntu.def
```

Note: Make sure you have a *access token*, otherwise the build will fail.

After building, you can test your container like so:

```
$ ./ubuntu.sif
hello world from ubuntu container!
```

You can also use the web GUI to build containers remotely. First, go to https://cloud.sylabs.io/builder (make sure you are signed in). Then you can copy and paste, upload, or type your definition file. When you are finished, click build. Then you can download the container with the URL.

CHAPTER

FIVE

ADVANCED USAGE

Once you've understood the basics, explore all the options which SingularityCE provides for accessing data, running persistent services in containers, manipulating the container environment, and applying networking and security configuration.

5.1 Bind Paths and Mounts

Unless disabled by the system administrator, SingularityCE allows you to map directories on your host system to directories within your container using bind mounts. This allows you to read and write data on the host system with ease.

5.1.1 Overview

When SingularityCE 'swaps' the host operating system for the one inside your container, the host file systems becomes inaccessible. However, you may want to read and write files on the host system from within the container. To enable this functionality, SingularityCE will bind directories back into the container via two primary methods: system-defined bind paths and user-defined bind paths.

5.1.2 System-defined bind paths

The system administrator has the ability to define which bind paths will be included automatically inside each container. Some bind paths are derived (e.g. a user's home directory), and some are statically defined with a `bind path` entry in `singularity.conf`.

In the default configuration, the default bind mounts are:

- The user's home directory (`$HOME`)
- The current working directory (`CWD`), unless its path contains symlinks resolving to different locations on the host vs inside the container.
- `/dev`
- `/etc/hosts`
- `/etc/localtime`
- `/proc`
- `/sys`
- `/tmp`
- `/var/tmp`

Additionally, customized system configuration files `/etc/resolv.conf`, `/etc/group`, and `/etc/passwd` are mounted into the container. These are adapted at container startup to match the requested configuration of the container.

Disabling System Binds

The `--no-mount` flag allows specific system mounts to be disabled, even if they are set in the `singularity.conf` configuration file by the administrator.

For example, if SingularityCE has been configured with `mount hostfs = yes` then every filesystem on the host will be bind mounted to the container by default. If, e.g. a `/project` filesystem on your host conflicts with a `/project` directory in the container you are running, you can disable the `hostfs` binds:

```
$ singularity run --no-mount hostfs mycontainer.sif
```

Multiple system mounts can be disabled by specifying them separated by commas:

```
$ singularity run --no-mount tmp,sys,dev mycontainer.sif
```

When the administrator has configured custom `bind path` entries in `singularity.conf`, to mount specific paths into the container by default, you can disable them individually. To do this, specify the path(s) to disable with the `--no-mount` flag. For example, if the adminstrator has configured `singularity.conf` to always mount `/data2`. you can disable this with `--no-mount /data2`:

```
$ singularity run --no-mount /data2 mycontainer.sif
```

To disable all `bind path` entries set in `singularity.conf`, use `--no-mount bind-paths`:

```
$ singularity run --no-mount bind-paths mycontainer.sif
```

5.1.3 User-defined bind paths

Unless the system administrator has disabled user control of binds, you will be able to request your own bind paths within your container.

The SingularityCE action commands (`run`, `exec`, `shell`, and `instance start`) will accept the `--bind/-B` command-line option to specify bind paths, and will also honor the `$SINGULARITY_BIND` (or `$SINGULARITY_BINDPATH`) environment variable. The argument for this option is a comma-delimited string of bind path specifications in the format `src[:dest[:opts]]`, where `src` and `dest` are paths outside and inside of the container respectively. If `dest` is not given, it is set equal to `src`. Mount options (`opts`) may be specified as `ro` (read-only) or `rw` (read/write, which is the default). The `--bind/-B` option can be specified multiple times, or a comma-delimited string of bind path specifications can be used.

SingularityCE 3.9 adds an additional `--mount` flag, which provides a longer-form method of specifying binds in `--mount type=bind,src=<source>,dst=<destination>[,<option>]...` format. This is compatible with the `--mount` syntax for binds in Docker and other OCI runtimes.

`--bind` Examples

Here's an example of using the `--bind` option and binding `/data` on the host to `/mnt` in the container (`/mnt` does not need to already exist in the container):

```
$ ls /data
bar  foo

$ singularity exec --bind /data:/mnt my_container.sif ls /mnt
bar  foo
```

You can bind multiple directories in a single command with this syntax:

```
$ singularity shell --bind /opt,/data:/mnt my_container.sif
```

This will bind `/opt` on the host to `/opt` in the container and `/data` on the host to `/mnt` in the container.

Using the environment variable instead of the command line argument, this would be:

```
$ export SINGULARITY_BIND="/opt,/data:/mnt"

$ singularity shell my_container.sif
```

Using the environment variable `$SINGULARITY_BIND`, you can bind paths even when you are running your container as an executable file with a runscript. If you bind many directories into your SingularityCE containers and they don't change, you could even benefit by setting this variable in your `.bashrc` file.

`--mount` Examples

The `--mount` flag takes a mount specification in the format `type=bind,src=<source>,dst=<dest>`. Additional options can be specified, comma delimited.

SingularityCE only supports the `bind` type for `--mount`, and will infer `type=bind` if it is not provided.

`src` or `source` can be used interchangeably. `dst`, `destination`, or `target` are also equivalent.

To mount `data` on the host to `/mnt` inside the container:

```
$ singularity exec \
    --mount type=bind,src=/data,dst=/mnt \
    my_container.sif ls /mnt
bar  foo
```

To mount the same directory read-only in the container, add the `ro` option:

```
$ singularity exec \
    --mount type=bind,source=/data,dest=/mnt,ro \
    my_container.sif touch /mnt/test
touch: cannot touch '/mnt/test': Permission denied
```

You can bind multiple directories in a single command with multiple `--mount` flags:

```
$ singularity shell --mount type=bind,src=/opt,dst=/opt \
                    --mount type=bind,src=/data,dst=/data \
                    my_container.sif
```

This will bind /opt on the host to /opt in the container and /data on the host to /mnt in the container.

The mount string can be quoted and escaped according to CSV rules, wrapping each field in double quotes if necessary characters. --mount allows bind mounting paths that are not possible with the --bind flag. For example:

```
# Mount a path containing ':' (not possible with --bind)
$ singularity run \
    --mount type=bind,src=/my:path,dst=/mnt \
    mycontainer.sif

# Mount a path containing a ','
$ singularity run \
    --mount type=bind,"src=/comma,dir",dst=/mnt \
    mycontainer.sif
```

Mount specifications are also read from then environment variable $SINGULARITY_MOUNT. Multiple bind mounts set via this environment variable should be separated by newlines (\n).

Using --bind or --mount with the --writable flag

To mount a bind path inside the container, a *bind point* must be defined within the container. The bind point is a directory within the container that SingularityCE can use as a destination to bind a directory on the host system.

Starting in version 3.0, SingularityCE will do its best to bind mount requested paths into a container regardless of whether the appropriate bind point exists within the container. SingularityCE can often carry out this operation even in the absence of the "overlay fs" feature.

However, binding paths to non-existent points within the container can result in unexpected behavior when used in conjunction with the --writable flag, and is therefore disallowed. If you need to specify bind paths in combination with the --writable flag, please ensure that the appropriate bind points exist within the container. If they do not already exist, it will be necessary to modify the container and create them.

Using --no-home and --containall flags

--no-home

When starting a container, SingularityCE allows you to mount your current working directory (CWD) without mounting your host $HOME directory by using the --no-home flag. This is equivalent to --no-mount home:

```
$ singularity shell --no-home my_container.sif

-or-

$ singularity shell --no-mount home my_container.sif
```

Disabling the mount of $HOME may be useful if your container image has files at $HOME, which would otherwise be hidden by the bind mount from the host.

Note: If your current working directory is under $HOME, and you do not want to mount it, you will need to disable both cwd and home mounts:

```
$ singularity shell --no-mount home,cwd my_container.sif
```

--contain / --containall

When using the `--contain` / `--containall` (or `-c` / `-C` for short) flags, `$HOME` from the host is not mounted, and an in-memory temporary directory is created at the `$HOME` point inside the container instead. An in-memory temporary directory is also used for `/tmp` and `/var/tmp` inside the container.

If the container image includes files within `$HOME`, the mounted temporary directory will hide them unless you also specify `--no-home` or `--no-mount home`:

```
$ singularity shell --containall my_container.sif
Singularity> ls -lah $HOME
total 4K
drwxr-xr-x    2 user group      60 Sep  1 11:45 .
drwxr-xr-x    1 user group      60 Sep  1 11:44 ..

$ singularity shell --containall --no-home my_container.sif
Singularity> ls -lah $HOME
total 52K
drwxr-xr-x    2 user group        60 Sep  1 11:45 .
drwxr-xr-x    1 user group        60 Sep  1 11:44 ..
drwxr-xr-x    1 user group     38672 Sep  1 11:44 mydata.csv
drwxr-xr-x    1 user group     14235 Sep  1 11:44 myimage.jpg
```

5.1.4 FUSE mounts

Filesystem in Userspace (FUSE) is an interface to allow filesystems to be mounted using code that runs in userspace, rather than in the Linux Kernel. Unprivileged (non-root) users can mount filesystems that have FUSE drivers. For example, the `fuse-sshfs` package allows you to mount a remote computer's filesystem to your local host, over ssh:

```
$ mount.fuse sshfs#ythel:/home/dave other_host/

# Now mounted to my local machine:
$ ythel:/home/dave on /home/dave/other_host type fuse.sshfs (rw,nosuid,nodev,relatime,
↪user_id=1000,group_id=1000)
```

SingularityCE 3.6 introduces the `--fusemount` option, which allows you directly expose FUSE filesystems inside a container. The FUSE command / driver that mounts a particular type of filesystem can be located on the host, or in the container.

Note: `--fusemount` functionality was present in a hidden preview state from SingularityCE 3.4. The behavior has changed for the final supported version introduced in SingularityCE 3.6.

Requirements

The FUSE command *must* be based on libfuse3 3.3.0 or greater to work correctly with SingularityCE. Older versions do not support the way in which the SingularityCE runtime passes a pre-mounted file descriptor into the container.

If you are using an older distribution that provides FUSE commands such as `sshfs` based on FUSE 2 then you can install FUSE 3 versions of the commands you need inside your container. EL7 distributions can install a compatible version of FUSE 3 from the EPEL repository. EL8 distributions ship FUSE 3.2.1 as a base package. Unfortunately this is an older version which does not fully support the way in which SingularityCE prepares FUSE mounts.

FUSE mount definitions

A fusemount definition for SingularityCE consists of 3 parts:

```
--fusemount <type>:<fuse command> <container mountpoint>
```

- **type** specifies how and where the FUSE mount will be run. The options are:
  - `host` - use a FUSE command on the host, to mount a filesystem into the container, with the fuse process attached.
  - `container` - use a FUSE command inside the container, to mount a filesystem into the container, with the fuse process attached.
  - `host-daemon` - use a FUSE command on the host, to mount a filesystem into the container, with the fuse process detached.
  - `container-daemon` - use a FUSE command inside the container, to mount a filesystem into the container, with the fuse process detached.
- **fuse command** specifies the name of the executable that implements the FUSE mount, and any arguments. E.g. `sshfs server:over-there/` for mounting a remote filesystem over SSH, where the remote source is `over-there/` in my home directory on the machine called `server`.
- **container mountpoint** is an *absolute path* at which the FUSE filesystem will be mounted in the container.

FUSE mount with a host executable

To use a FUSE `sshfs` mount in a container, where the `fuse-sshfs` package has been installed on my host, I run with the `host` mount type:

```
$ singularity run --fusemount "host:sshfs server:/ /server" docker://ubuntu
Singularity> cat /etc/hostname
localhost.localdomain
Singularity> cat /server/etc/hostname
server
```

FUSE mount with a container executable

If the FUSE driver / command that you want to use for the mount has been added to your container, you can use the `container` mount type:

```
$ singularity run --fusemount "container:sshfs server:/ /server" sshfs.sif
Singularity> cat /etc/hostname
localhost.localdomain
Singularity> cat /server/etc/hostname
server
```

5.1.5 Image Mounts

In SingularityCE 3.6 and above you can mount a directory contained in an image file into a container. This may be useful if you want to distribute directories containing a large number of data files as a single image file.

You can mount from image files in ext3 format, squashfs format, or SIF format.

The ext3 image file format allows you to mount it into the container read/write and make changes, while the other formats are read-only. Note that you can only use a read/write image in a single container. You cannot mount it to multiple container runs at the same time.

To mount a directory from an image file, use the `-B/--bind` option and specify the bind in the format:

```
-B <image-file>:<dest>:image-src=<source>
```

Alternatively use the `--mount` option, and specify the bind in the format:

```
--mount type=bind,src=<image-file>,dst=<dest>,image-src=<source>
```

This will bind the `<source>` path inside `<image-file>` to `<dest>` in the container.

If you do not add `:image-src=<source>` to your bind specification, then the `<image-file>` itself will be bound to `<dest>` instead.

Ext3 Image Files

If you have a directory called `inputs/` that holds data files you wish to distribute in an image file that allows read/write:

```
# Create an image file 'inputs.img' of size 100MB and put the
# files inputs/ into it's root directory
$ mkfs.ext3 -d inputs/ inputs.img 100M
mke2fs 1.45.6 (20-Mar-2020)
Creating regular file inputs.img
Creating filesystem with 102400 1k blocks and 25688 inodes
Filesystem UUID: e23c29c9-7a49-4b82-89bf-2faf36b5a781
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Copying files into the device: done
Writing superblocks and filesystem accounting information: done
```

(continues on next page)

(continued from previous page)

```
# Run SingularityCE, mounting my input data to '/input-data' in
# the container.
$ singularity run -B inputs.img:/input-data:image-src=/ mycontainer.sif
Singularity> ls /input-data
1           3           5           7           9
2           4           6           8           lost+found

# Or with --mount instead of -B
$ singularity run \
    --mount type=bind,src=inputs.img,dst=/input-data,image-src=/ \
    mycontainer.sif
```

SquashFS Image Files

If you have a directory called `inputs/` that holds data files you wish to distribute in an image file that is read-only, and compressed, then the squashfs format is appropriate:

```
# Create an image file 'inputs.squashfs' and put the files from
# inputs/ into it's root directory
$ mksquashfs inputs/ inputs.squashfs
Parallel mksquashfs: Using 16 processors
Creating 4.0 filesystem on inputs.squashfs, block size 131072.
...

# Run SingularityCE, mounting my input data to '/input-data' in
# the container.
$ singularity run -B inputs.squashfs:/input-data:image-src=/ mycontainer.sif
Singularity> ls /input-data/
1  2  3  4  5  6  7  8  9

# Or with --mount instead of -B
$ singularity run \
    --mount type=bind,src=src-inputs.squashfs,dst=/input-data,image-src=/ \
    mycontainer.sif
```

SIF Image Files

Advanced users may wish to create a standalone SIF image, which contains an `ext3` or `squashfs` data partition holding files, by using the `singularity sif` commands similarly to the *persistent overlays instructions*:

```
# Create a new empty SIF file
$ singularity sif new inputs.sif

# Add the squashfs data image from above to the SIF
$ singularity sif add --datatype 4 --partarch 2 --partfs 1 --parttype 3 inputs.sif␣
↪inputs.squashfs

# Run SingularityCE, binding data from the SIF file
$ singularity run -B inputs.sif:/input-data:image-src=/ mycontainer.sif
```

(continues on next page)

(continued from previous page)

```
Singularity> ls /input-data
1  2  3  4  5  6  7  8  9

# Or with --mount instead of -B
$ singularity run \
    --mount type=bind,src=inputs.sif,dst=/input-data,image-src=/ \
    mycontainer.sif
```

If your bind source is a SIF then SingularityCE will bind from the first data partition in the SIF, or you may specify an alternative descriptor by ID with the additional option `id=n`, where n is the descriptor ID.

5.2 Persistent Overlays

Persistent overlay directories allow you to overlay a writable file system on an immutable read-only container for the illusion of read-write access. You can run a container and make changes, and these changes are kept separately from the base container image.

5.2.1 Overview

A persistent overlay is a directory or file system image that "sits on top" of your immutable SIF container. When you install new software or create and modify files the overlay will store the changes.

If you want to use a SIF container as though it were writable, you can create a directory, an ext3 file system image, or embed an ext3 file system image in SIF to use as a persistent overlay. Then you can specify that you want to use the directory or image as an overlay at runtime with the `--overlay` option, or `--writable` if you want to use the overlay embedded in SIF.

If you want to make changes to the image, but do not want them to persist, use the `--writable-tmpfs` option. This stores all changes in an in-memory temporary filesystem which is discarded as soon as the container finishes executing.

Note: The `--writable-tmpfs` size is controlled by `sessiondir max size` in `singularity.conf`. This defaults to 64MiB, and may need to be increased if your workflows create larger temporary files.

You can use persistent overlays with the following commands:

- `run`
- `exec`
- `shell`
- `instance start`

5.2.2 Usage

To use a persistent overlay, you must first have a container.

```
$ sudo singularity build ubuntu.sif library://ubuntu
```

Filesystem image overlay

Since 3.8, SingularityCE provides a command `singularity overlay create` to create persistent overlay images.

Note: `dd` and `mkfs.ext3` must be installed on your system. Additionally `mkfs.ext3` must support `-d` option in order to create an overlay directory tree usable by a regular user.

For example, to create a 1 GiB overlay image:

```
$ singularity overlay create --size 1024 /tmp/ext3_overlay.img
```

`singularity overlay create` also provides an option `--create-dir` to create additional directories owned by the calling user. This option can be specified multiple times to create several such directories. This is particularly useful when you need to make a directory that is writable by your user.

For example:

```
$ singularity build /tmp/nginx.sif docker://nginx
$ singularity overlay create --size 1024 --create-dir /var/cache/nginx /tmp/nginx_
↪overlay.img
$ echo "test" | singularity exec --overlay /tmp/nginx_overlay.img /tmp/nginx.sif sh -c
↪"cat > /var/cache/nginx/test"
```

Note: Filesystem image overlays are only supported when {singularity} is installed in setuid mode. An unprivileged installation of SingularityCE can create these kinds of overlays, but cannot mount them to the container at runtime.

Sparse overlay images

Since 3.11, SingularityCE allows the creation of overlay images as sparse files. A sparse overlay image only takes up space on disk as data is written to it. A standard overlay image will use an amount of disk space equal to its size, from the time that it is created.

To create a sparse overlay image, use the `--sparse` flag.

```
$ singularity overlay create --sparse --size 1024 /tmp/ext3_overlay.img
```

Note that `ls` will show the full size of the file, while `du` will show the space on disk that the file is currently using:

```
$ ls -lah /tmp/ext3_overlay.img
-rw-------. 1 dtrudg-sylabs dtrudg-sylabs 1.0G Jan 27 11:47 /tmp/ext3_overlay.img

$ du -h /tmp/ext3_overlay.img
33M     /tmp/ext3_overlay.img
```

If you copy or move the sparse image you should ensure that the tool you use to do so supports sparse files, which may require enabling an option. Failure to copy or move the file with sparse file support will lead to it taking its full size on disk in the new location.

Create an overlay image manually

You can use tools like `dd` and `mkfs.ext3` to create and format an empty ext3 file system image that will be used as an overlay.

To create an overlay image file with 500MBs of empty space:

```
$ dd if=/dev/zero of=overlay.img bs=1M count=500 && \
    mkfs.ext3 overlay.img
```

Now you can use this overlay with your container, though filesystem permissions still control where you can write, so `sudo` is needed to run the container as `root` if you need to write to `/` inside the container.

```
$ sudo singularity shell --overlay overlay.img ubuntu.sif
```

To manage permissions in the overlay, so the container is writable by unprivileged users you can create a directory structure on your host, set permissions on it as needed, and include it in the overlay with the `-d` option to `mkfs.ext3`:

```
$ mkdir -p overlay/upper overlay/work
$ dd if=/dev/zero of=overlay.img bs=1M count=500 && \
     mkfs.ext3 -d overlay overlay.img
```

Now the container will be writable as the unprivileged user who created the `overlay/upper` and `overlay/work` directories that were placed into `overlay.img`.

```
$ singularity shell --overlay overlay.img ubuntu.sif
Singularity> echo $USER
dtrudg
Singularity> echo "Hello" > /hello
```

Note: The `-d` option to `mkfs.ext3` does not support `uid` or `gid` values >65535. To allow writes from users with larger uids you can create the directories for your overlay with open permissions, e.g. `mkdir -p -m 777 overlay/upper overlay/work`. At runtime files and directories created in the overlay will have the correct `uid` and `gid`, but it is not possible to lock down permissions so that the overlay is only writable by certain users.

Directory overlay

A directory overlay is simpler to use than a filesystem image overlay, but a directory of modifications to a base container image cannot be transported or shared as easily as a single overlay file.

Note: For security reasons, if SingularityCE is installed in setuid mode, you must be root to use a bare directory as an overlay. ext3 file system images can be used as overlays without root privileges.

Non-root users can use directory overlays if SingularityCE is installed in non-setuid mode, and the kernel (>=5.11) of the system supports this.

Create a directory as usual:

```
$ mkdir my_overlay
```

The example below shows the directory overlay in action.

```
$ sudo singularity shell --overlay my_overlay/ ubuntu.sif

SingularityCE ubuntu.sif:~> mkdir /data

SingularityCE ubuntu.sif:~> chown user /data

SingularityCE ubuntu.sif:~> apt-get update && apt-get install -y vim

SingularityCE ubuntu.sif:~> which vim
/usr/bin/vim

SingularityCE ubuntu.sif:~> exit
```

Overlay embedded in SIF

It is possible to embed an overlay image into the SIF file that holds a container. This allows the read-only container image and your modifications to it to be managed as a single file.

To add a 1 GiB writable overlay partition to an existing SIF image:

```
$ singularity overlay create --size 1024 ubuntu.sif
```

Warning: It is not possible to add a writable overlay partition to a **signed**, **encrypted** SIF image or if the SIF image already contains a writable overlay partition.

`singularity overlay create` also provides an option `--create-dir` to create additional directories owned by the calling user, it can be specified multiple times to create many directories. This is particularly useful when you need to make a directory writable by your user.

So for example:

```
$ singularity build /tmp/nginx.sif docker://nginx
$ singularity overlay create --size 1024 --create-dir /var/cache/nginx /tmp/nginx.sif
$ echo "test" | singularity exec /tmp/nginx.sif sh -c "cat > /var/cache/nginx/test"
```

Note: SIF embedded overlays are only supported when {singularity} is installed in setuid mode. An unprivileged installation of SingularityCE can create these kinds of overlays, but cannot mount them to the container at runtime.

Embed an overlay image in SIF

To embed an existing overlay in a SIF image, or to create an empty overlay when using SingularityCE <3.8, use the `sif add` subcommand.

In order to do this, you must first create a file system image:

```
$ dd if=/dev/zero of=overlay.img bs=1M count=500 && \
    mkfs.ext3 overlay.img
```

Then, you can add the overlay to the SIF image using the `sif` functionality of SingularityCE.

```
$ singularity sif add --datatype 4 --partfs 2 --parttype 4 --partarch 2 --groupid 1␣
↪ubuntu_latest.sif overlay.img
```

Below is the explanation what each parameter means, and how it can possibly affect the operation:

- `datatype` determines what kind of an object we attach, e.g. a definition file, environment variable, signature.
- `partfs` should be set according to the partition type, e.g. SquashFS, ext3, raw.
- `parttype` determines the type of partition. In our case it is being set to overlay.
- `partarch` must be set to the architecture against which you're building. In this case it's `amd64`.
- `groupid` is the ID of the container image group. In most cases there's no more than one group, therefore we can assume it is 1.

All of these options are documented within the CLI help. Access it by running `singularity sif add --help`.

After you've completed the steps above, you can shell into your container with the `--writable` option.

```
$ sudo singularity shell --writable ubuntu_latest.sif
```

Final note

You will find that when using the `--overlay` option, your changes persist across sessions as though you were using a writable container.

```
$ singularity shell --overlay my_overlay/ ubuntu.sif

SingularityCE ubuntu.sif:~> ls -lasd /data
4 drwxr-xr-x 2 user root 4096 Apr  9 10:21 /data

SingularityCE ubuntu.sif:~> which vim
/usr/bin/vim

SingularityCE ubuntu.sif:~> exit
```

If you mount your container without the `--overlay` directory, your changes will be gone.

```
$ singularity shell ubuntu.sif

SingularityCE ubuntu.sif:~> ls /data
ls: cannot access 'data': No such file or directory

SingularityCE ubuntu.sif:~> which vim
```

(continues on next page)

(continued from previous page)

```
SingularityCE ubuntu.sif:~> exit
```

To resize an overlay, standard Linux tools which manipulate ext3 images can be used. For instance, to resize the 500MB file created above to 700MB one could use the `e2fsck` and `resize2fs` utilities as follows:

```
$ e2fsck -f my_overlay && \
    resize2fs my_overlay 700M
```

More information on creating and manipulating ext3 images on various Linux distribution are available where documentation for those respective distributions is found.

5.3 Instances - Running Services

SingularityCE is most commonly used to run containers interactively, or in a batch job, where the container runs in the foreground, performs some work, and then exits. There are *different ways* in which you can run SingularityCE containers in the foreground. If you use `run`, `exec` and `shell` to interact with processes in the container, then you are running SingularityCE containers in the foreground.

SingularityCE, also allows you to run containers in a "detached" or "daemon" mode where the container runs a service. A "service" is essentially a process running in the background that multiple different clients can use. For example, a web server or a database.

A SingularityCE container running a service in the background is called an *instance*, to distinguish it from the default mode which runs containers in the foreground.

5.3.1 Overview

This page will help you understand instances using an elementary example followed by a more useful example running an NGINX web server using instances. At the end, you will find a more detailed example of running an instance of a service exposing an API that converts URL to PDFs.

To run a service, such as a web server, outside of a container you would typically install the package for the web server and then instruct systemd (which manages system services on most Linux distributions) to start it. E.g.

```
$ sudo yum install nginx
$ sudo systemctl enable --now nginx
```

If you were to attempt this in a container, it's likely that it will not work as expected. You cannot use systemd to run services in a container, by default. It expects various conditions that are satisfied on the host system, but not inside containers.

Instead, you should run NGINX in the foreground *inside* the container, but then run the container in the background, as an *instance*.

5.3.2 Container Instances in SingularityCE

To demonstrate the basices of instances, let's use an easy (though somewhat useless) example, using an Alpine Linux image from the container library:

```
$ singularity pull library://alpine
```

The above command will save the alpine image from the Container Library as `alpine_latest.sif`.

Starting Instances

To start an instance, you should follow this procedure :

```
[command]                       [image]              [name of instance]

$ singularity instance start   alpine_latest.sif     instance1
```

This command causes SingularityCE to create an isolated environment for the container services to live inside. You can confirm that an instance is running by using the `instance list` command:

```
$ singularity instance list

INSTANCE NAME    PID      IP              IMAGE
instance1        22084                    /home/dave/instances/alpine_latest.sif
```

Note: Instances are linked to the user account that started them. This means that if you use `sudo` to start an instance as `root`, you will need to use `sudo` for all commands managing that instance. `instance list` will not show instances started by other users.

If you want to run multiple instances from the same image, it's as simple as running the command multiple times with different instance names. The instance name uniquely identify instances, so they cannot be repeated.

```
$ singularity instance start alpine_latest.sif instance2

$ singularity instance start alpine_latest.sif instance3
```

We now have 3 instances, all using the same image:

```
$ singularity instance list

INSTANCE NAME    PID      IP              IMAGE
instance1        22084                    /home/dave/instances/alpine_latest.sif
instance2        22443                    /home/dave/instances/alpine_latest.sif
instance3        22493                    /home/dave/instances/alpine_latest.sif
```

You can filter the instance list by supplying a pattern:

```
$ singularity instance list '*2'

INSTANCE NAME    PID      IP              IMAGE
instance2        22443                    /home/dave/instances/alpine_latest.s
```

When an instance is started, it will begin to run the `%startscript` from the container's *definition file* in the background. If there is no `%startscript` the container will stay idle in the background.

Interacting With Instances

Although an instance runs its `%startscript` (if there is one) in the background, you can also interact with it in the foreground, by referring to it with an `instance://<name>` URI, where `<name>` is replaced with the instance name.

To run a specific command against an instance, in the foreground, use `singularity exec`:

```
$ singularity exec instance://instance1 cat /etc/os-release
```

Similarly, you can use `singularity run` to run the `%runscript` for the container, against a running instance:

```
$ singularity run instance://instance2
```

If you want to poke around inside of your instance, you can use the normal `singularity shell` command, but give it the instance URI:

```
$ singularity shell instance://instance3

Singularity>
```

Stopping Instances

When you are finished with your instance you can clean it up with the `instance stop` command as follows:

```
$ singularity instance stop instance1
```

If you have multiple instances running and you want to stop all of them, you can do so with a wildcard or the –all flag. The following three commands are identical.

```
$ singularity instance stop \*

$ singularity instance stop --all

$ singularity instance stop -a
```

Note: You must escape the wildcard with a backslash * to pass it properly through your shell.

5.3.3 Nginx "Hello-world" in SingularityCE

The above example, although not very useful, should serve as a fair introduction to the concept of SingularityCE instances and running containers in the background. We will now look at a more useful example of setting up an NGINX web server using instances. First we will create a basic *definition file* (let's call it nginx.def):

```
Bootstrap: docker
From: nginx

%startscript
   nginx
```

This downloads the official NGINX Docker container, converts it to a SingularityCE image, and tells it to run the `nginx` command when you start the instance. Because we are running a web server, which defaults to listening on privileged port 80, we're going to run the following instance commands as root, using `sudo`.

```
$ sudo singularity build nginx.sif nginx.def

$ sudo singularity instance start --writable-tmpfs nginx.sif web
```

The `--writable-tmpfs` option is needed, because NGINX will attempt to write some files when it starts up. `--writable-tmpfs` allows these to be written to a temporary, in-memory location, that will be removed when the instance is stopped.

Just like that we've downloaded, built, and run an NGINX SingularityCE image. We can confirm it's running using the curl tool, to fetch the web page that is now being hosted by NGINX.

```
$ curl localhost

<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
 body {
     width: 35em;
     margin: 0 auto;
     font-family: Tahoma, Verdana, Arial, sans-serif;
 }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

You could also visit `http://localhost` in a web browser, if you are running the instance from a desktop session.

5.3.4 API Server Example

Let's now package a useful service into a SIF container, and run it as an instance. The service we will be packaging is an API server that converts a web page into a PDF, and can be found here.

Building the image

To package the Web to PDF service into a SIF container, we must create a definition file. Let's first choose a base from which to build our container. In this case the docker image `node:8` which comes pre-installed with Node 8 has been used:

```
Bootstrap: docker
From: node:8
```

The service also requires a slew of dependencies to be manually installed in addition to Node 8, so we can add those into the `post` section as well as calling the installation script for the `url-to-pdf`:

```
%post

    apt-get update && apt-get install -yq gconf-service libasound2 \
        libatk1.0-0 libc6 libcairo2 libcups2 libdbus-1-3 libexpat1 \
        libfontconfig1 libgcc1 libgconf-2-4 libgdk-pixbuf2.0-0 \
        libglib2.0-0 libgtk-3-0 libnspr4 libpango-1.0-0 \
        libpangocairo-1.0-0 libstdc++6 libx11-6 libx11-xcb1 libxcb1 \
        libxcomposite1 libxcursor1 libxdamage1 libxext6 libxfixes3 libxi6 \
        libxrandr2 libxrender1 libxss1 libxtst6 ca-certificates \
        fonts-liberation libappindicator1 libnss3 lsb-release xdg-utils \
        wget curl && rm -r /var/lib/apt/lists/*
    git clone https://github.com/alvarcarto/url-to-pdf-api.git pdf_server
    cd pdf_server
    npm install
    touch .env
    chmod -R 0755 .
```

We need to define what happens when we start an instance of the container by writing a `%startscript`. In this situation, we want to run the commands that start up the url-to-pdf service:

```
%startscript
    cd /pdf_server
    npm start
```

Also, the `url-to-pdf` service requires some environment variables to be set, which we can do in the environment section:

```
%environment
    NODE_ENV=development
    PORT=9000
    ALLOW_HTTP=true
    URL=localhost
    export NODE_ENV PORT ALLOW_HTTP URL
```

The complete definition file will look like this:

```
Bootstrap: docker
From: node:8

%post

    apt-get update && apt-get install -yq gconf-service libasound2 \
        libatk1.0-0 libc6 libcairo2 libcups2 libdbus-1-3 libexpat1 \
        libfontconfig1 libgcc1 libgconf-2-4 libgdk-pixbuf2.0-0 \
        libglib2.0-0 libgtk-3-0 libnspr4 libpango-1.0-0 \
        libpangocairo-1.0-0 libstdc++6 libx11-6 libx11-xcb1 libxcb1 \
        libxcomposite1 libxcursor1 libxdamage1 libxext6 libxfixes3 libxi6 \
        libxrandr2 libxrender1 libxss1 libxtst6 ca-certificates \
        fonts-liberation libappindicator1 libnss3 lsb-release xdg-utils \
        wget curl && rm -r /var/lib/apt/lists/*
    git clone https://github.com/alvarcarto/url-to-pdf-api.git pdf_server
    cd pdf_server
    npm install
    touch .env
    chmod -R 0755 .

%startscript
    cd /pdf_server
    npm start

%environment
    NODE_ENV=development
    PORT=9000
    ALLOW_HTTP=true
    URL=localhost
    export NODE_ENV PORT ALLOW_HTTP URL
```

We can now build the container image from the definition file:

```
$ sudo singularity build url-to-pdf.sif url-to-pdf.def
```

Running the Service

We can now start an instance to run the service:

```
$ singularity instance start url-to-pdf.sif pdf
```

Because the service listens on port 9000, which is not a privileged port, we don't need to run it with sudo this time.

We can confirm it's working by sending the server an http request using curl:

```
$ curl -o sylabs.pdf localhost:9000/api/render?url=http://sylabs.io/docs

% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                             Dload  Upload   Total   Spent    Left  Speed

100 73750  100 73750    0     0  14583      0  0:00:05  0:00:05 --:--:-- 19130
```

You should see a PDF file being generated like the one shown below:

If you shell into the instance, you can see the processes that are running, to provide the service:

```
$ singularity shell instance://pdf
Singularity> ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
dtrudg-+       1  0.0  0.0 1178984 20700 ?        Sl   11:40   0:00 sinit
dtrudg-+      13  0.0  0.0   4284   696 ?        S    11:40   0:00 /bin/sh /.singularity.
↪d/startscript
dtrudg-+      15  1.0  0.0 984908 41508 ?        Sl   11:40   0:00 npm
dtrudg-+      25  0.0  0.0   4292   716 ?        S    11:40   0:00 sh -c env-cmd nodemon␣
↪--watch ./src -e js src/index.js
dtrudg-+      26  0.1  0.0 876908 31084 ?        Sl   11:40   0:00 node /pdf_server/node_
↪modules/.bin/env-cmd nodemon --watch ./src -e js src/index
dtrudg-+      32  0.7  0.0 1113984 39976 ?       Sl   11:40   0:00 node /pdf_server/node_
↪modules/.bin/nodemon --watch ./src -e js src/index.js
dtrudg-+      44  1.7  0.0 941556 53804 ?        Sl   11:40   0:00 /usr/local/bin/node␣
↪src/index.js
dtrudg-+     124  0.0  0.0  18372  3592 pts/1    S    11:41   0:00 /bin/bash --norc
dtrudg-+     130  0.0  0.0  36640  2836 pts/1    R+   11:41   0:00 ps aux
```

5.3.5 Instance Logs

Generally, when running services using instances, we write the `%startscript` so that the service will run in the foreground, and would write any log messages to the terminal. When an instance container is started there is no terminal. SingularityCE moves the container into the background, and collects output and error messages into log files.

You can view the location of log files for running instances using the `--log` option of the `instance list` command:

```
$ singularity instance list --logs
INSTANCE NAME    PID       LOGS
pdf              935864    /home/dtrudg-sylabs/.singularity/instances/logs/mini/dtrudg-
↪sylabs/pdf.err
                           /home/dtrudg-sylabs/.singularity/instances/logs/mini/dtrudg-
↪sylabs/pdf.out
```

Note that the log files are located under `.singularity/instances` in the user's home directory, and are grouped by the hostname, and instance name.

The `.out` log collects standard output. The `.err` log collects standard error. You can look at the content of the log files to check how your service is running:

```
$ cat /home/dtrudg_sylabs/.singularity/instances/logs/mini/dtrudg-sylabs/pdf.out

> url-to-pdf-api@1.0.0 start /pdf_server
> env-cmd nodemon --watch ./src -e js src/index.js

[nodemon] 1.19.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: /pdf_server/src/**/*
[nodemon] starting `node src/index.js`
2023-02-01T11:14:58.185Z - info: [app.js] ALLOW_HTTP=true, unsafe requests are allowed.␣
↪Don't use this in production.
2023-02-01T11:14:58.187Z - info: [app.js] ALLOW_URLS set! Allowed urls patterns are:
2023-02-01T11:14:58.187Z - info: [app.js] Using CORS options: origin=*, methods=[GET,␣
↪POST, PUT, DELETE, OPTIONS, HEAD, PATCH]
2023-02-01T11:14:58.206Z - warn: [router.js] Warning: no authentication required to use␣
↪the API
2023-02-01T11:14:58.209Z - info: [index.js] Express server listening on http://
↪localhost:9000/ in development mode
2023-02-01T11:15:17.269Z - info: [render-core.js] Rendering with opts: {
...
```

5.3.6 Resource Usage / Limits

If you are running a container as the `root` user, or your system supports cgroups v2, then all instances will be started inside a cgroup. A cgroup allows the resources used by the instance to be monitored, and limited.

To monitor the resource usage of an instance, use the `instance stats` command:

```
 $ singularity instance stats pdf
INSTANCE NAME    CPU USAGE    MEM USAGE / LIMIT     MEM %    BLOCK I/O            PIDS
pdf              0.00%        479.8MiB / 62.2GiB    0.75%    470MiB / 131.6MiB    45
```

We can see that the instance is currently idle (0.00% CPU), and is using 479.8MiB of RAM. No limits have been applied, so the total RAM size of the machine is shown.

By default, `instance stats` is interactive when run from a terminal, and will update every second. To obtain point-in-time usage details use the `--no-stream` or `--json` options.

Where supported by the system's cgroups configuration, resource limits can be applied to instances using the same *command line flags* that are available for interactive containers. E.g. to limit memory usage to 1GiB, we can use the `--memory` flag:

```
singularity instance start --memory 1G url-to-pdf.sif pdf
```

5.3.7 System integration / PID files

If you are running services in containers you may want them to be started on boot, and shutdown gracefully automatically. This is usually performed by an init process, or another supervisor daemon installed on your host. Many init and supervisor daemons support managing processes via pid files.

You can specify a `--pid-file` option to `singularity instance start` to write the PID for an instance to the specified file, e.g.

```
$ singularity instance start --pid-file /home/dave/alpine.pid alpine_latest.sif instanceA

$ cat /home/dave/alpine.pid
23727
```

An example service file for an instance controlled by systemd is below. This can be used as a template to setup containerized services under systemd.

```
[Unit]
Description=Web Instance
After=network.target

[Service]
Type=forking
Restart=always
User=www-data
Group=www-data
PIDFile=/run/web-instance.pid
ExecStart=/usr/local/bin/singularity instance start --pid-file /run/web-instance.pid /
↪data/containers/web.sif web-instance
ExecStop=/usr/local/bin/singularity instance stop web-instance

[Install]
WantedBy=multi-user.target
```

Note that `Type=forking` is required here, since `instance start` starts an instance and then exits.

5.4 Environment and Metadata

Environment variables are values you can set in a session, which can be used to influence the behavior of programs. It's often considered best practice to use environment variables to pass settings to a program in a container, because they are easily set and don't rely on writing and binding in program-specific configuration files. When building a container you may need to set fixed or default environment variables. When running containers you may need to set or override environment variables.

The *metadata* of a container is information that describes the container. SingularityCE automatically records important information such as the definition file used to build a container. Other details such as the version of SingularityCE used are present as *labels* on a container. You can also specify your own to be recorded against your container.

5.4.1 Environment Overview

When you run a program in a container with SingularityCE, the environment variables that the program sees are a combination of:

- The environment variables set in the base image (e.g. Docker image) used to build the container.
- The environment variables set in the `%environment` section of the definition file used to build the container.
- *Most* of the environment variables set on your host, which are passed into the container.
- Any variables you set specifically for the container at runtime, using the `--env`, `--env-file` options, or by setting `SINGULARITYENV_` variables outside of the container.
- The `PATH` variable can be manipulated to add entries.
- Runtime variables `SINGULARITY_xxx` set by SingularityCE to provide information about the container.

The environment variables from the base image or definition file used to build a container always apply, but can be overridden.

You can choose to exclude passing environment variables from the host into the container with the `-e` or `--cleanenv` option.

We'll go through each place environment variables can be defined, so that you can understand how the final environment in a container is created, and can be manipulated.

If you are interested in variables available when you are *building* a container, rather than when running a container, see *build environment section*.

5.4.2 Environment From a Base Image

When you build a container with SingularityCE you might *bootstrap* from a library or Docker image, or using Linux distribution bootstrap tools such as `debootstrap`, `yum` etc.

When using `debootstrap`, `yum` etc. you are starting from a fresh install of a Linux distribution into your container. No specific environment variables will be set. If you are using a `library` or `Docker` source then you may inherit environment variables from your base image.

If I build a singularity container from the image `docker://python:3.7` then when I run the container I can see that the `PYTHON_VERSION` variable is set in the container:

```
$ singularity exec python.sif env | grep PYTHON_VERSION
PYTHON_VERSION=3.7.7
```

This happens because the `Dockerfile` used to build that container has `ENV PYTHON_VERSION 3.7.7` set inside it.

You can override the inherited environment with `SINGULARITYENV_` vars, or the `--env / --env-file` flags (see below), but `Dockerfile` `ENV` vars will not be overridden by host environment variables of the same name.

5.4.3 Environment From a Definition File

Environment variables can be included in your container by adding them to your definition file. Use `export` in the `%environment` section of a definition file to set a container environment variable:

```
Bootstrap: library
From: default/alpine

%environment
    export MYVAR="Hello"

%runscript
    echo $MYVAR
```

Now the value of `MYVAR` is `Hello` when the container is launched. The `%runscript` is set to echo the value.

```
$ singularity run env.sif
Hello
```

Warning: SingularityCE 3.6 uses an embedded shell interpreter to evaluate and setup container environments, therefore all commands executed from the `%environment` section have an execution timeout of **5 seconds** for SingularityCE 3.6 and a **1 minute** timeout since SingularityCE 3.7. While it is possible to source a script from there, it is not recommended to use this section to run potentially long initialization tasks because this would impact users running the image and the execution could abort due to timeout.

Build Time Variables in `%post`

In some circumstances the value that needs to be assigned to an environment variable may only be known after e.g. software installation, in `%post`. For situations like this, the `$SINGULARITY_ENVIRONMENT` variable is provided. Redirecting text to this variable will cause it to be written to a file called `/.singularity.d/env/91-environment.sh` that will be sourced at runtime.

Variables set in the `%post` section through `$SINGULARITY_ENVIRONMENT` take precedence over those added via `%environment`.

5.4.4 Environment From the Host

If you have environment variables set outside of your container, on the host, then by default they will be available inside the container. Except that:

- An environment variable set on the host will be overridden by a variable of the same name that has been set either inside the container image, or via `SINGULARITYENV_` environment variables, or the `--env` and `--env-file` flags.
- The `PS1` shell prompt is reset for a container specific prompt.
- The `PATH` environment variable will be modified to contain default values.

- The LD_LIBRARY_PATH is modified to a default /.singularity.d/libs, that will include NVIDIA / ROCm libraries if applicable.

To override an environment variable that is already set in the container with the value from the host, use SINGULARITY_ENV or the --env flag. For example, to force MYVAR in the container to take the value of MYVAR on the host:

```
$ export SINGULARITYENV_MYVAR="$MYVAR"
$ singularity run mycontainer.sif

# or
$ singularity run --env "MYVAR=$MYVAR"
```

If you *do not want* the host environment variables to pass into the container you can use the -e or --cleanenv option. This gives a clean environment inside the container, with a minimal set of environment variables for correct operation of most software.

```
$ singularity exec --cleanenv env.sif env
HOME=/home/dave
LANG=C
LD_LIBRARY_PATH=/.singularity.d/libs
PATH=/startpath:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
PROMPT_COMMAND=PS1="Singularity> "; unset PROMPT_COMMAND
PS1=Singularity>
PWD=/home/dave/doc-tesrts
SINGULARITY_COMMAND=exec
SINGULARITY_CONTAINER=/home/dave/doc-tesrts/env.sif
SINGULARITY_ENVIRONMENT=/.singularity.d/env/91-environment.sh
SINGULARITY_NAME=env.sif
TERM=xterm-256color
```

Warning: If you work on a host system that sets a lot of environment variables, e.g. because you use software made available through environment modules / lmod, you may see strange behavior in your container. Check your host environment with env for variables such as PYTHONPATH that can change the way code runs, and consider using --cleanenv.

5.4.5 Environment From the SingularityCE Runtime

It can be useful for a program to know when it is running in a SingularityCE container, and some basic information about the container environment. SingularityCE will automatically set a number of environment variables in a container that can be inspected by any program running in the container.

- SINGULARITY_COMMAND - how the container was started, e.g. exec / run / shell.
- SINGULARITY_CONTAINER - the full path to the container image.
- SINGULARITY_ENVIRONMENT - path inside the container to the shell script holding the container image environment settings.
- SINGULARITY_NAME - name of the container image, e.g. myfile.sif or docker://ubuntu.
- SINGULARITY_BIND - a list of bind paths that the user requested, via flags or environment variables, when running the container.

5.4.6 Overriding Environment Variables

You can override variables that have been set in the container image, or define additional variables, in various ways as appropriate for your workflow.

`--env` option

The `--env` option on the `run/exec/shell` commands allows you to specify environment variables as `NAME=VALUE` pairs:

```
$ singularity run env.sif
Hello

$ singularity run --env MYVAR=Goodbye env.sif
Goodbye
```

Separate multiple variables with commas, e.g. `--env MYVAR=A,MYVAR2=B`, and use shell quoting / shell escape if your variables include special characters.

`--env-file` option

The `--env-file` option lets you provide a file that contains environment variables as `NAME=VALUE` pairs, e.g.:

```
$ cat myenvs
MYVAR="Hello from a file"

$ singularity run --env-file myenvs env.sif
Hello from a file
```

`SINGULARITYENV_` prefix

If you export an environment variable on your host called `SINGULARITYENV_xxx` *before* you run a container, then it will set the environment variable `xxx` inside the container:

```
$ singularity run env.sif
Hello

$ export SINGULARITYENV_MYVAR="Overridden"
$ singularity run env.sif
Overridden
```

Manipulating `PATH`

`PATH` is a special environment variable that tells a system where to look for programs that can be run. `PATH` contains multiple filesystem locations (paths) separated by colons. When you ask to run a program `myprog`, the system looks through these locations one by one, until it finds `myprog`.

To ensure containers work correctly, when a host `PATH` might contain a lot of host-specific locations that are not present in the container, SingularityCE will ensure `PATH` in the container is set to a default.

```
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

This covers the standard locations for software installed using a system package manager in most Linux distributions. If you have software installed elsewhere in the container, then you can override this by setting `PATH` in the container definition `%environment` block.

If your container depends on things that are bind mounted into it, or you have another need to modify the `PATH` variable when starting a container, you can do so with `SINGULARITYENV_APPEND_PATH` or `SINGULARITYENV_PREPEND_PATH`.

If you set a variable on your host called `SINGULARITYENV_APPEND_PATH` then its value will be appended (added to the end) of the `PATH` variable in the container.

```
$ singularity exec env.sif sh -c 'echo $PATH'
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

$ export SINGULARITYENV_APPEND_PATH="/endpath"
$ singularity exec env.sif sh -c 'echo $PATH'
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/endpath
```

Alternatively you could use the `--env` option to set a `APPEND_PATH` variable, e.g. `--env APPEND_PATH=/endpath`.

If you set a variable on your host called `SINGULARITYENV_PREPEND_PATH` then its value will be prepended (added to the start) of the `PATH` variable in the container.

```
$ singularity exec env.sif sh -c 'echo $PATH'
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

$ export SINGULARITYENV_PREPEND_PATH="/startpath"
$ singularity exec env.sif sh -c 'echo $PATH'
/startpath:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

Alternatively you could use the `--env` option to set a `PREPEND_PATH` variable, e.g. `--env PREPEND_PATH=/startpath`.

5.4.7 Escaping and Evaluation of Environment Variables

In its default native mode, SingularityCE uses an embedded shell interpreter to process the container startup scripts and environment. When this processing is performed, by default a single step of shell evaluation happens in the container context. The shell from which you are running SingularityCE may also evaluate variables on your command line before passing them to SingularityCE.

In OCI-mode (`--oci`), SingularityCE follows OCI behavior and does not shell evaluate environment.

Docker / OCI Compatibility

This default behavior of SingularityCE, in its native mode, differs from Docker/OCI handling of environment variables / `ENV` directives. To avoid the extra evaluation of variables that SingularityCE performs you can:

- Follow the instructions about escaping in the sections below, to add additional escape characters and/or quoting.
- Use the `--no-eval` or `--compat` flags.

Note: When running a container in OCI-mode (`--oci`), SingularityCE follows Docker/OCI behaviour by default. You do not need to enable the `--no-eval` or `--compat` options.

`--no-eval` prevents SingularityCE from evaluating environment variables on container startup, so that they will take the same value as with a Docker/OCI runtime:

```
# Set an environment variable that would run `date` if evaluated
$ export SINGULARITYENV_MYVAR='$(date)'

# Default behavior
# MYVAR was evaluated in the container, and is set to the output of `date`
$ singularity run ~/ubuntu_latest.sif env | grep MYVAR
MYVAR=Tue Apr 26 14:37:07 CDT 2022

# --no-eval / --compat behavior
# MYVAR was not evaluated and is a literal `$(date)`
$ singularity run --no-eval ~/ubuntu_latest.sif env | grep MYVAR
MYVAR=$(date)
```

The `--compat` flag is a short-hand flag to activate `--no-eval` along with other Docker/OCI compatibility flags. See *Docker-like --compat Flag* for more details.

Using Host Variables

To set a container environment variable to the value of a variable on the host, use double quotes around the variable, so that it is processed by the host shell before the value is passed to SingularityCE. For example:

```
singularity run --env "MYHOST=$HOSTNAME" mycontainer.sif
```

This will set the `MYHOST` environment variable inside the container to the value of the `HOSTNAME` on the host system. `$HOSTNAME` is substituted before the host shell runs `singularity`.

Note: You can often use no quotes, but it is good practice to use quotes consistently so that variables containing e.g. spaces are handled correctly.

Using Container Variables

To set an environment variable to a value that references another variable inside the container, you should escape the `$` sign to `\$`. This prevents the host shell from substituting the value. Instead it will be substituted inside the container.

For example, to create an environment variable `MYPATH`, with the same value as `PATH` in the container (not the host's `PATH`):

```
singularity run --env "MYPATH=\$PATH" mycontainer.sif
```

You can also use this approach to append or prepend to variables that are already set in the container. For example, `--env PATH="\$PATH:/endpath"` would have the same effect as `--env APPEND_PATH="/endpath"`, which uses the special `APPEND/PREPEND` handling for `PATH` discussed above.

Quoting / Avoiding Evaluation

If you need to pass an environment variable into the container verbatim, it must be quoted and escaped appropriately. For example, if you need to set a path containing a literal `$LIB` for the `LD_PRELOAD` environment variable:

```
singularity run --env="LD_PRELOAD=/foo/bar/\\\$LIB/baz.so" mycontainer.sif
```

This will result in `LD_PRELOAD` having the value `/foo/bar/$LIB/baz.so` inside the container.

The host shell consumes the double `\\`, and then environment processing within SingularityCE will consume the third `\` that escapes the literal `$`.

You can also use single quotes on the command line, to avoid one level of escaping:

```
singularity run --env='LD_PRELOAD=/foo/bar/\$LIB/baz.so' mycontainer.sif
```

5.4.8 Environment Variable Precedence

When a container is run with SingularityCE, the container environment is constructed in the following order:

- Clear the environment, keeping just `HOME` and `SINGULARITY_APPNAME`.
- Set Docker/OCI defined environment variables, where a Docker or OCI image was used as the base for the container build.
- If `PATH` is not defined set the SingularityCE default `PATH` *or*
- If `PATH` is defined, add any missing path parts from SingularityCE defaults
- Set environment variables defined explicitly in the `%environment` section of the definition file. These can override any previously set values.
- Set environment variables that were defined in the `%post` section of the build, by addition to the `$SINGULARITY_ENVIRONMENT` file.
- Set SCIF (`--app`) environment variables
- Set base environment essential vars (`PS1` and `LD_LIBRARY_PATH`)
- Inject `SINGULARITYENV_` / `--env` / `--env-file` variables so they can override or modify any previous values.
- Apply special `APPEND_PATH` / `PREPEND_PATH` handling.
- Restore environment variables from the host, if they have not already been set in the container, and the `--cleanenv` / `--containall` options were not specified.

Warning: While SingularityCE will process additional scripts found under `/.singularity.d/env` inside the container, it is strongly recommended to avoid manipulating the container environment by directly adding or modifying scripts in this directory. Please use the `%environment` section of the definition file, and the `$SINGULARITY_ENVIRONMENT` file from `%post` if required.

A future version of SingularityCE may move container scripts, environment, and metadata outside of the container's root filesystem. This will permit further reproducibility and compatibility improvements, but will preclude environment manipulation via arbitrary scripts.

5.4.9 Umask / Default File Permissions

The `umask` value on a Linux system controls the default permissions for newly created files. It is not an environment variable, but influences the behavior of programs in the container when they create new files.

Note: A detailed description of what the `umask` is, and how it works can be found at Wikipedia.

SingularityCE 3.7 and above set the `umask` in the container to match the value outside, unless:

- The `--fakeroot` option is used, in which case a `0022` umask is set so that `root` owned newly created files have expected 'system default' permissions, and can be accessed by other non-root users who may use the same container later.
- The `--no-umask` option is used, in which case a `0022` umask is set.

Note: In SingularityCE 3.6 and below a default `0022` umask was always applied.

5.4.10 Container Metadata

Each SingularityCE container has metadata describing the container, how it was built, etc. This metadata includes the definition file used to build the container and labels, which are specific pieces of information set automatically or explicitly when the container is built.

For containers that are generated with SingularityCE version 3.0 and later, default labels are represented using the rc1 Label Schema.

Inherited Labels

When building a container from an existing image, either directly from a URI or with a definition file, your container will inherit the labels that are set in that base image. For example the `LABEL` a Docker container sets in its `Dockerfile`, or a SIF container that sets labels in its definition file as described below.

Inherited labels can only be overwritten during a build when the build is performed using the `--force` option. SingularityCE will warn that it is not modifying an existing label when `--force` is not used:

```
$ singularity build test2.sif test2.def
...
INFO:    Adding labels
WARNING: Label: OWNER already exists and force option is false, not overwriting
```

Note: SingularityCE 3.0 through 3.8 did not inherit labels from Docker/OCI images during a build. SingularityCE 3.9 restores the behavior of 2.x, and inherits these labels.

Custom Labels

You can add custom labels to your container using the `%labels` section in a definition file:

```
Bootstrap: library
From: ubuntu:latest

%labels
  OWNER Joana
```

Dynamic Build Time Labels

You may wish to set a label to a value that is not known in advance, when you are writing the definition file, but can be obtained in the `%post` section of your definition file while the container is building.

SingularityCE 3.7 and above allow this, through adding labels to the file defined by the `SINGULARITY_LABELS` environment variable in the `%post` section:

```
Bootstrap: library
From: ubuntu:latest

# These labels take a fixed value in the definition
%labels
  OWNER Joana

# We can now also set labels to a value at build time
%post
  VAL="$(myprog --version)"
  echo "my.label $VAL" >> "$SINGULARITY_LABELS"
```

Labels must be added to the file one per line, in a `NAME VALUE` format, where the name and value are separated by a space.

Inspecting Metadata

The `inspect` command gives you the ability to view the labels and/or other metadata that were added to your container when it was built.

-l/ --labels

Running inspect without any options, or with the `-l` or `--labels` options will display any labels set on the container

```
$ singularity inspect ubuntu.sif
my.label: version 1.2.3
OWNER: Joana
org.label-schema.build-arch: amd64
org.label-schema.build-date: Thursday_12_November_2020_10:51:59_CST
org.label-schema.schema-version: 1.0
org.label-schema.usage.singularity.deffile.bootstrap: library
org.label-schema.usage.singularity.deffile.from: ubuntu:latest
org.label-schema.usage.singularity.version: 3.7.0-rc.1
```

We can easily see when the container was built, the source of the base image, and the exact version of SingularityCE that was used to build it.

The custom label `OWNER` that we set in our definition file is also visible.

-d / --deffile

The `-d` or `-deffile` flag shows the definition file(s) that were used to build the container.

```
$ singularity inspect --deffile jupyter.sif
```

And the output would look like:

```
Bootstrap: library
From: debian:9

%help
    Container with Anaconda 2 (Conda 4.5.11 Canary) and Jupyter Notebook 5.6.0 for␣
↪Debian 9.x (Stretch).
    This installation is based on Python 2.7.15

%environment
    JUP_PORT=8888
    JUP_IPNAME=localhost
    export JUP_PORT JUP_IPNAME

%startscript
    PORT=""
    if [ -n "$JUP_PORT" ]; then
    PORT="--port=${JUP_PORT}"
    fi

    IPNAME=""
    if [ -n "$JUP_IPNAME" ]; then
    IPNAME="--ip=${JUP_IPNAME}"
    fi

    exec jupyter notebook --allow-root ${PORT} ${IPNAME}

%setup
    #Create the .condarc file where the environments/channels from conda are specified,␣
↪these are pulled with preference to root
    cd /
    touch .condarc

%post
    echo 'export RANDOM=123456' >>$SINGULARITY_ENVIRONMENT
    #Installing all dependencies
    apt-get update && apt-get -y upgrade
    apt-get -y install \
    build-essential \
    wget \
    bzip2 \
```

(continues on next page)

(continued from previous page)

```
    ca-certificates \
    libglib2.0-0 \
    libxext6 \
    libsm6 \
    libxrender1 \
    git
    rm -rf /var/lib/apt/lists/*
    apt-get clean
    #Installing Anaconda 2 and Conda 4.5.11
    wget -c https://repo.continuum.io/archive/Anaconda2-5.3.0-Linux-x86_64.sh
    /bin/bash Anaconda2-5.3.0-Linux-x86_64.sh -bfp /usr/local
    #Conda configuration of channels from .condarc file
    conda config --file /.condarc --add channels defaults
    conda config --file /.condarc --add channels conda-forge
    conda update conda
    #List installed environments
    conda list
```

Which is the definition file for the `jupyter.sif` container.

-r / --runscript

The `-r` or `--runscript` option shows the runscript for the image.

```
$ singularity inspect --runscript jupyter.sif
```

And the output would look like:

```
#!/bin/sh
OCI_ENTRYPOINT=""
OCI_CMD="bash"
# ENTRYPOINT only - run entrypoint plus args
if [ -z "$OCI_CMD" ] && [ -n "$OCI_ENTRYPOINT" ]; then
SINGULARITY_OCI_RUN="${OCI_ENTRYPOINT} $@"
fi

# CMD only - run CMD or override with args
if [ -n "$OCI_CMD" ] && [ -z "$OCI_ENTRYPOINT" ]; then
if [ $# -gt 0 ]; then
    SINGULARITY_OCI_RUN="$@"
else
    SINGULARITY_OCI_RUN="${OCI_CMD}"
fi
fi

# ENTRYPOINT and CMD - run ENTRYPOINT with CMD as default args
# override with user provided args
if [ $# -gt 0 ]; then
    SINGULARITY_OCI_RUN="${OCI_ENTRYPOINT} $@"
else
    SINGULARITY_OCI_RUN="${OCI_ENTRYPOINT} ${OCI_CMD}"
```

(continues on next page)

(continued from previous page)

```
fi

exec $SINGULARITY_OCI_RUN
```

-t / --test

The `-t` or `--test` flag shows the test script for the image.

```
$ singularity inspect --test jupyter.sif
```

This will output the corresponding `%test` section from the definition file.

-e / --environment

The `-e` or `--environment` flag shows the environment variables that are defined in the container image. These may be set from one or more environment files, depending on how the container was built.

```
$ singularity inspect --environment jupyter.sif
```

And the output would look like:

```
==90-environment.sh==
#!/bin/sh

JUP_PORT=8888
JUP_IPNAME=localhost
export JUP_PORT JUP_IPNAME
```

-H / --helpfile

The `-H` or `-helpfile` flag will show the container's description in the `%help` section of its definition file.

You can call it this way:

```
$ singularity inspect --helpfile jupyter.sif
```

And the output would look like:

```
Container with Anaconda 2 (Conda 4.5.11 Canary) and Jupyter Notebook 5.6.0 for Debian 9.
↪x (Stretch).
This installation is based on Python 2.7.15
```

-j / --json

This flag gives you the possibility to output your labels in a JSON format.

You can call it this way:

```
$ singularity inspect --json ubuntu.sif
```

And the output would look like:

```
{
        "data": {
                "attributes": {
                        "labels": {
                                "my.label": "version 1.2.3",
                                "OWNER": "Joana",
                                "org.label-schema.build-arch": "amd64",
                                "org.label-schema.build-date": "Thursday_12_November_
↪2020_10:51:59_CST",
                                "org.label-schema.schema-version": "1.0",
                                "org.label-schema.usage.singularity.deffile.bootstrap":
↪"library",
                                "org.label-schema.usage.singularity.deffile.from":
↪"ubuntu:latest",
                                "org.label-schema.usage.singularity.version": "3.7.0-rc.1
↪"
                        }
                }
        },
        "type": "container"
}
```

5.4.11 /.singularity.d directory

The `/.singularity.d` directory in a container contains scripts and environment files that are used when a container is executed.

You should not manually modify files under `/.singularity.d`, from your definition file during builds, or directly within your container image. Recent 3.x versions of SingularityCE replace older action scripts dynamically, at runtime, to support new features. In the longer term, metadata will be moved outside of the container, and stored only in the SIF file metadata descriptor.

```
/.singularity.d/

├── actions
│   ├── exec
│   ├── run
│   ├── shell
│   ├── start
│   └── test
├── env
│   ├── 01-base.sh
│   ├── 10-docker2singularity.sh
```

(continues on next page)

(continued from previous page)

```
│   ├── 90-environment.sh
│   ├── 91-environment.sh
│   ├── 94-appsbase.sh
│   ├── 95-apps.sh
│   └── 99-base.sh
├── labels.json
├── libs
├── runscript
├── runscript.help
├── Singularity
└── startscript
```

- **actions**: This directory contains helper scripts to allow the container to carry out the action commands. (e.g. `exec` , `run` or `shell`). In later versions of SingularityCE, these files may be dynamically written at runtime, *and should not be modified* in the container.
- **env**: All `*.sh` files in this directory are sourced in alphanumeric order when the container is started. For legacy purposes there is a symbolic link called `/environment` that points to `/.singularity.d/env/90-environment.sh`. Whenever possible, avoid modifying or creating environment files manually to prevent potential issues building & running containers with future versions of SingularityCE. Additional facilities such as `--env` and `--env-file` are available to allow manipulation of the container environment at runtime.
- **labels.json**: The json file that stores a containers labels described above.
- **libs**: At runtime the user may request some host-system libraries to be mapped into the container (with the `--nv` option for example). If so, this is their destination.
- **runscript**: The commands in this file will be executed when the container is invoked with the `run` command or called as an executable. For legacy purposes there is a symbolic link called `/singularity` that points to this file.
- **runscript.help**: Contains the description that was added in the `%help` section.
- **SingularityCE**: This is the definition file that was used to generate the container. If more than 1 definition file was used to generate the container additional SingularityCE files will appear in numeric order in a sub-directory called `bootstrap_history`.
- **startscript**: The commands in this file will be executed when the container is invoked with the `instance start` command.

5.5 Plugins

5.5.1 Overview

A SingularityCE plugin is a package that can be dynamically loaded by the SingularityCE runtime, augmenting SingularityCE with experimental, non-standard and/or vendor-specific functionality.

Plugins can influence the behaviour of SingularityCE in specific ways:

- A cli plugin can use the `Command` callback to add or modify CLI subcommands and/or flags.
- A cli plugin can use the `SingularityEngineConfig` callback to change the container configuration before it is passed to the runtime, e.g. adding bind mounts etc.
- A runtime plugin can use the `MonitorContainer` callback to watch the container process as it is executing.
- A runtime plugin can use the `PostStartProcess` callback to carry out a task after the container has been started.

- A runtime plugin can use the `RegisterImageDriver` callback to implement an alternative way of providing a container image to execute.

5.5.2 Limitations / Requirements

The way that plugin functionality is implemented in the Go language, which SingularityCE is written with, is quite restrictive.

Go plugins must be built with the same Go version, and set of dependencies, as the main program they will be loaded into. This means it is generally impractical to develop and build plugins except in lock-step with the main SingularityCE source tree.

Functionality that can be implemented with plugins is limited to the scope of the exposed plugin callbacks. Container runtimes such as SingularityCE execute using multiple processes, with distinct boundaries that limit the influence a plugin can have.

If you are considering writing a plugin for SingularityCE you may wish to investigate whether the feature can be contributed to the main source tree directly via a PR. This simplifies future maintenance, and avoids the limitations of Go plugins.

5.5.3 Using Plugins

The `list` command prints the currently installed plugins.

```
$ singularity plugin list
There are no plugins installed.
```

Plugins are packaged and distributed as binaries encoded with the versatile Singularity Image Format (SIF). However, plugin authors may also distribute the source code of their plugins. A plugin can be compiled from its source code with the `compile` command. A number of example plugins are included in the `examples/plugins` directory of the SingularityCE source.

```
$ singularity plugin compile examples/plugins/cli-plugin/
INFO:    Plugin built to: /home/dtrudg/Git/singularity/examples/plugins/cli-plugin/cli-
↪plugin.sif
```

Upon successful compilation, a SIF file will appear in the directory of the plugin's source code.

```
$ ls examples/plugins/cli-plugin/ | grep sif
cli-plugin.sif
```

Note: Due to the structure of the SingularityCE project, and the strict requirements of Go plugin compilation, **all** plugins must be compiled from within the SingularityCE source code tree.

The ability to compile plugins outside of the SingularityCE tree, that previously existed, has been removed due to incompatible changes in Go 1.18.

Every plugin encapsulates various information such as the plugin's author, the plugin's version, etc. To view this information about a plugin, use the `inspect` command.

```
$ singularity plugin inspect examples/plugins/cli-plugin/cli-plugin.sif
Name: github.com/sylabs/singularity/cli-example-plugin
Description: This is a short example CLI plugin for Singularity
```

(continues on next page)

(continued from previous page)

```
Author: Sylabs Team
Version: 0.1.0
```

To compile a plugin, use the `compile` command.

```
$ singularity plugin compile examples/plugins/log-plugin/
INFO:    Plugin built to: /home/myuser/singularity/examples/plugins/log-plugin/log-
↪plugin.sif
```

Note: Before using the `plugin compile` subcommand, make sure that you trust the origin of the plugin, and that you are certain it does not contain any malicious code.

To install a plugin, use the `install` command. This operation requires root privilege.

```
$ sudo singularity plugin install examples/plugins/cli-plugin/cli-plugin.sif
$ singularity plugin list
ENABLED  NAME
    yes  sylabs.io/cli-plugin
```

Note: Before using the `plugin install` subcommand, make sure that you trust the origin of the plugin, and that you are certain it does not contain any malicious code.

For further information on verifying the contents of SIF files using cryptographic signatures, see the *Sign and Verify section*.

After successful installation, the plugin will automatically be enabled. Any plugin can be disabled with the `disable` command and re-enabled with the `enable` command. Both of these operations require root privilege.

```
$ sudo singularity plugin disable sylabs.io/cli-plugin
$ singularity plugin list
ENABLED  NAME
     no  sylabs.io/cli-plugin

$ sudo singularity plugin enable sylabs.io/cli-plugin
$ singularity plugin list
ENABLED  NAME
    yes  sylabs.io/cli-plugin
```

Finally, to uninstall a plugin, use the `uninstall` command. This operation requires root privilege.

```
$ sudo singularity plugin uninstall sylabs.io/cli-plugin
Uninstalled plugin "sylabs.io/cli-plugin".

$ singularity plugin list
There are no plugins installed.
```

5.5.4 Writing a Plugin

Developers interested in writing SingularityCE plugins can get started by reading the Go documentation for the plugin package.

Example plugins can be found in the SingularityCE source code.

5.6 Security Options

SingularityCE can make use of various Linux kernel features to modify the security scope and context of running containers. Non-root users may be granted additional permissions using Linux capabilities. SELinux, AppArmor, and Seccomp can be used to restrict the operations that can be performed by a container.

5.6.1 Linux Capabilities

Native runtime / non-OCI-Mode

In SingularityCE's default configuration, without `--oci`, a container started by root receives all capabilities, while a container started by a non-root user receives no capabilities.

Additionally, SingularityCE provides support for granting and revoking Linux capabilities on a user or group basis. For example, let us suppose that an administrator has decided to grant a user (named `pinger`) capabilities to open raw sockets so that they can use `ping` in a container where the binary is controlled via capabilities. For information about how to manage capabilities as an admin please refer to the capability admin docs.

Note: In SingularityCE's default setuid and non-OCI mode, containers are only isolated in a mount namespace. A user namespace, which limits the scope of capabilities, is not used by default.

Therefore, it is extremely important to recognize that **granting users Linux capabilities with the `capability` command group is usually identical to granting those users root level access on the host system**. Most, if not all, capabilities will allow users to "break out" of the container and become root on the host. This feature is targeted toward special use cases (like cloud-native architectures) where an admin/developer might want to limit the attack surface within a container that normally runs as root. This is not a good option in multi-tenant HPC environments where an admin wants to grant a user special privileges within a container. For that and similar use cases, the *fakeroot feature* is a better option.

To take advantage of this granted capability as a user, `pinger` must also request the capability when executing a container with the `--add-caps` flag like so:

```
$ singularity exec --add-caps CAP_NET_RAW library://sylabs/tests/ubuntu_ping:v1.0 ping -
↪c 1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=52 time=73.1 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 73.178/73.178/73.178/0.000 ms
```

If the admin decides that it is no longer necessary to allow the user `pinger` to open raw sockets within SingularityCE containers, they can revoke the appropriate Linux capability and `pinger` will not be able to add that capability to their containers anymore:

```
$ singularity exec --add-caps CAP_NET_RAW library://sylabs/tests/ubuntu_ping:v1.0 ping -
↪c 1 8.8.8.8
WARNING: not authorized to add capability: CAP_NET_RAW
ping: socket: Operation not permitted
```

Another scenario which is atypical of shared resource environments, but useful in cloud-native architectures is dropping capabilities when spawning containers as the root user to help minimize attack surfaces. With a default installation of SingularityCE, containers created by the root user will maintain all capabilities. This behavior is configurable if desired. Check out the capability configuration and root default capabilities sections of the admin docs for more information.

Assuming the root user will execute containers with the `CAP_NET_RAW` capability by default, executing the same container `pinger` executed above works without the need to grant capabilities:

```
# singularity exec library://sylabs/tests/ubuntu_ping:v1.0 ping -c 1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=52 time=59.6 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 59.673/59.673/59.673/0.000 ms
```

Now we can manually drop the `CAP_NET_RAW` capability like so:

```
# singularity exec --drop-caps CAP_NET_RAW library://sylabs/tests/ubuntu_ping:v1.0 ping -
↪c 1 8.8.8.8
ping: socket: Operation not permitted
```

And now the container will not have the ability to create new sockets, causing the `ping` command to fail.

The `--add-caps` and `--drop-caps` options will accept the `all` keyword. Of course appropriate caution should be exercised when using this keyword.

OCI-Mode

When containers are run in OCI-mode, by a non-root user, initialization is always performed inside a user namespace. The capabilities granted to a container are specific to this user namespace. For example, `CAP_SYS_ADMIN` granted to an OCI-mode container does not give the user the ability to mount a filesystem outside of the container's user namespace.

Because of this isolation of capabilities users can add and drop capabilities, using `--add-caps` and `--drop-caps`, without the need for the administrator to have granted permission to do so with the `singularity capabilities` command.

OCI-mode containers do not inherit the user's own capabilities, but instead run with a default set of capabilities that matches other OCI runtimes.

- CAP_NET_RAW
- CAP_NET_BIND_SERVICE
- CAP_AUDIT_READ
- CAP_AUDIT_WRITE
- CAP_DAC_OVERRIDE
- CAP_SETFCAP
- CAP_SETPCAP

- CAP_SETGID
- CAP_SETUID
- CAP_MKNOD
- CAP_CHOWN
- CAP_FOWNER
- CAP_FSETID
- CAP_KILL
- CAP_SYS_CHROOT

When the container is entered as the root user (e.g. with `--fakeroot`), these default capabilities are added to the effective, permitted, and bounding sets.

When the container is entered as a non-root user, these default capabilities are added to the bounding set.

5.6.2 Security related action options

When starting a container with the action commands `shell`, `exec`, and `run`, various flags allow fine grained control of security.

`--add-caps`

In the default non-OCI-mode, `--add-caps` will grant specified Linux capabilities (e.g. `CAP_NET_RAW`) to a container, provided that those capabilities have been granted to the user by an administrator using the `capability add` command. This option will also accept the case insensitive keyword `all` to add every capability granted by the administrator.

In OCI-mode, `--add-caps` will grant specified Linux capabilities (e.g. `CAP_NET_RAW`) to the container. Because the container runs in a user namespace, the capabilities are not effective on the host and do not have to be granted by the administrator. The keyword `all` will grant all available capabilities to the container.

`--drop-caps`

In the default non-OCI-mode, the root user has a full set of capabilities when they enter the container. You may choose to drop specific capabilities when you initiate a container as root to enhance security.

For instance, to drop the ability for the root user to open a raw socket inside the container:

```
$ sudo singularity exec --drop-caps CAP_NET_RAW library://centos ping -c 1 8.8.8.8
ping: socket: Operation not permitted
```

In OCI-mode any user can use `--drop-caps` to run a container with fewer capabilities than the default OCI capability set.

The `--drop-caps` option will also accept the case insensitive keyword `all` as an option to drop all capabilities when entering the container.

--allow-setuid

The SetUID bit allows a program to be executed as the user that owns the binary. The most well-known SetUID binaries are owned by root and allow a user to execute a command with elevated privileges. But other SetUID binaries may allow a user to execute a command as a service account.

By default SetUID is disallowed within SingularityCE containers as a security precaution, by mounting container filesystems as `nosetuid`.

In the default non-OCI-mode, the root user can override this precaution and allow SetUID binaries to behave as expected within a SingularityCE container with the `--allow-setuid` option like so:

```
$ sudo singularity shell --allow-setuid some_container.sif
```

In OCI-mode, any user can permit SetUID binaries with the `--allow-setuid` option. Because an OCI-mode container is always run in a user namespace, SetUID will change to UIDs inside a user's permitted subuid/subgid mapping. This does not allow access to arbitrary UIDs on the host system.

--keep-privs

In the default non-OCI-mode, it is possible for an admin to set a different set of default capabilities or to reduce the default capabilities to zero for the root user by setting the `root default capabilities` parameter in the `singularity.conf` file to `file` or `no` respectively. If this change is in effect, the root user can override the `singularity.conf` file and enter the container with full capabilities using the `--keep-privs` option.

```
$ sudo singularity exec --keep-privs library://centos ping -c 1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=18.8 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 18.838/18.838/18.838/0.000 ms
```

In OCI-mode, the `--keep-privs` option can be used by any user. In this mode, `--keep-privs` will cause the container to run inheriting the current effective capabilities rather than using the OCI default capability set. When entering the container as a non-root user, the capabilities are only inherited to the bounding set.

--no-privs

In the default non-OCI-mode, the `--no-privs` option allows the root user to run a container with all capabilities dropped, and sets the `no_new_privs` bit that will prevent the container process gaining any further privilege.

In OCI-mode, the `--no-privs` option can be used by any user to run a container with all capabilities dropped, and to set the `no_new_privs` bit that will prevent the container process gaining any further privilege.

--security

The `--security` flag, currently supported in non-OCI-mode only, allows the root user to leverage security modules such as SELinux, AppArmor, and seccomp within your SingularityCE container. It is also possible to change the UID and GID of the user within the container at runtime.

For instance:

```
$ sudo whoami
root

$ sudo singularity exec --security uid:1000 my_container.sif whoami
david
```

To use seccomp to blacklist a command follow this procedure. (It is actually preferable from a security standpoint to whitelist commands but this will suffice for a simple example.) Note that this example was run on Ubuntu and that SingularityCE was installed with the `libseccomp-dev` and `pkg-config` packages as dependencies.

First write a configuration file. An example configuration file is installed with SingularityCE, normally at `/usr/local/etc/singularity/seccomp-profiles/default.json`. For this example, we will use a much simpler configuration file to blacklist the `mkdir` command.

```
{
    "defaultAction": "SCMP_ACT_ALLOW",
    "archMap": [
        {
            "architecture": "SCMP_ARCH_X86_64",
            "subArchitectures": [
                "SCMP_ARCH_X86",
                "SCMP_ARCH_X32"
            ]
        }
    ],
    "syscalls": [
        {
            "names": [
                "mkdir"
            ],
            "action": "SCMP_ACT_KILL",
            "args": [],
            "comment": "",
            "includes": {},
            "excludes": {}
        }
    ]
}
```

We'll save the file at `/home/david/no_mkdir.json`. Then we can invoke the container like so:

```
$ sudo singularity shell --security seccomp:/home/david/no_mkdir.json my_container.sif

Singularity> mkdir /tmp/foo
Bad system call (core dumped)
```

Note that attempting to use the blacklisted `mkdir` command resulted in a core dump.

The full list of arguments accepted by the `--security` option are as follows:

```
--security="seccomp:/usr/local/etc/singularity/seccomp-profiles/default.json"
--security="apparmor:/usr/bin/man"
--security="selinux:context"
--security="uid:1000"
--security="gid:1000"
--security="gid:1000:1:0" (multiple gids, first is always the primary group)
```

5.6.3 Encrypted containers

Beginning in SingularityCE 3.4.0 it is possible to build and run encrypted containers. The containers are decrypted at runtime entirely in kernel space, meaning that no intermediate decrypted data is ever present on disk. See *encrypted containers* for more details.

5.7 Network virtualization

SingularityCE provides full integration with cni , to make network virtualization easy. The following options can be used with the the action commands (`exec`, `run`, and `shell`) to create and configure a virtualized network for a container.

Note: Many of these options are available only to the `root` user by default. Unrestricted ability to configure networking for containers would allow users to affect networking on the host, or in a cluster.

SingularityCE 3.8 allows the administrator to permit a list of unprivileged users and/or groups to use specified network configurations. This is accomplished through settings in `singularity.conf`. See the administrator guide for details.

5.7.1 `--dns`

The `--dns` option allows you to specify a comma separated list of DNS servers to add to the `/etc/resolv.conf` file.

```
$ nslookup sylabs.io | grep Server
Server:             127.0.0.53

$ sudo singularity exec --dns 8.8.8.8 ubuntu.sif nslookup sylabs.io | grep Server
Server:             8.8.8.8

$ sudo singularity exec --dns 8.8.8.8 ubuntu.sif cat /etc/resolv.conf
nameserver 8.8.8.8
```

5.7.2 --hostname

The --hostname option accepts a string argument to change the hostname within the container.

```
$ hostname
ubuntu

$ sudo singularity exec --hostname hal-9000 my_container.sif hostname
hal-9000
```

5.7.3 --net

Passing the --net flag will cause the container to join a new network namespace when it initiates. New in SingularityCE 3.0, a bridge interface will also be set up by default.

```
$ hostname -I
10.0.2.15

$ sudo singularity exec --net my_container.sif hostname -I
10.22.0.4
```

5.7.4 --network

The --network option can only be invoked in combination with the --net flag. It accepts a comma delimited string of network types. Each entry will bring up a dedicated interface inside container.

```
$ hostname -I
172.16.107.251 10.22.0.1

$ sudo singularity exec --net --network ptp ubuntu.sif hostname -I
10.23.0.6

$ sudo singularity exec --net --network bridge,ptp ubuntu.sif hostname -I
10.22.0.14 10.23.0.7
```

When invoked, the --network option searches the singularity configuration directory (commonly /usr/local/etc/singularity/network/) for the cni configuration file corresponding to the requested network type(s). Several configuration files are installed with SingularityCE by default corresponding to the following network types:

- bridge
- ptp
- ipvlan
- macvlan
- none (must be used alone)

By default, none is the only network configuration that can be used by non-privileged users. It isolates the container network from the host network with a loopback interface.

Administrators can permit certain users or groups to request other network configurations through options in singularity.conf. Additional cni configuration files can be added to the network configuration directory as required, and SingularityCE's provided configurations may also be modified.

5.7.5 --network-args

The `--network-args` option provides a convenient way to specify arguments to pass directly to the cni plugins. It must be used in conjunction with the `--net` flag.

For instance, let's say you want to start an NGINX server on port 80 inside of the container, but you want to map it to port 8080 outside of the container:

```
$ sudo singularity instance start --writable-tmpfs \
    --net --network-args "portmap=8080:80/tcp" docker://nginx web2
```

The above command will start the Docker Hub official NGINX image running in a background instance called `web2`. The NGINX instance will need to be able to write to disk, so we've used the `--writable-tmpfs` argument to allocate some space in memory. The `--net` flag is necessary when using the `--network-args` option, and specifying the `portmap=8080:80/tcp` argument which will map port 80 inside of the container to 8080 on the host.

Now we can start NGINX inside of the container:

```
$ sudo singularity exec instance://web2 nginx
```

And the `curl` command can be used to verify that NGINX is running on the host port 8080 as expected.

```
$ curl localhost:8080
10.22.0.1 - - [16/Oct/2018:09:34:25 -0400] "GET / HTTP/1.1" 200 612 "-" "curl/7.58.0" "-"
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

For more information about cni, check the cni specification.

5.8 Limiting Container Resources

It's often useful to limit the resources that are consumed by a container, e.g. to allow the container to only use 1 CPU, or 50% of the RAM on the system. Although on HPC clusters it's common to launch containers with a job scheduler that can set limits per job, the following scenarios are examples where more direct control is useful:

- When running multiple containerized applications inside an HPC job, where each container in the job should have different resource limits.
- When testing HPC code on a workstation, to avoid excessive CPU / RAM usage bringing the desktop environment and other applications to a halt.
- When benchmarking code that doesn't provide internal means to limit the number of CPUs it uses.

There are three ways to apply limits to a container that is run with SingularityCE:

- Using the command line flags introduced in v3.10.
- Using the `--apply-cgroups` flag to apply a `cgroups.toml` file that defines the resource limits.
- Using external tools such as `systemd-run` tool to apply limits, and then call `singularity`.

5.8.1 Requirements - Linux Cgroups

Resource limits are applied to containers using functionality in the Linux kernel known as *control groups* or *cgroups*. There are two versions of *cgroups*:

cgroups v1 has a more complex structure, and allows only the root user to safely apply limits to applications. If your system is using cgroups v1 then you can only use the CLI resource limit flags or `--apply-cgroups` when running containers as the root user.

cgroups v2 has a simplified structure, and is designed in a way that permits *delegation* of cgroups control to standard users. This delegation is usually accomplished via *systemd*.

Generally, to apply resource limits to a container as a non-root user your system must:

- Be using cgroups v2, in the unified hierarchy mode.
- Have a Linux kernel version >= 4.15.
- Have systemd version >= 224.
- Have `systemd cgroups` enabled in `singularity.conf` (this is the default).
- Have systemd configured to delegate cgroups controllers to non-root users.

Recent distributions such as Ubuntu 22.04, Debian 11, Fedora 31, and newer, satisfy these criteria by default. On older distributions support can often be enabled. Consult the admin documentation or speak to your system administrator about this.

5.8.2 Command Line Limit Flags

SingularityCE 3.10 introduced a number of simple command line flags that you can use with *shell/run/exec* and the *instance* commands to directly apply resource limits to a container when you run it.

The flags detailed below are compatible with those used by the `docker` CLI, except that the short forms are not supported. For example, you cannot use `-c` instead of `--cpu-shares` because `-c` is used by SingularityCE for another purpose.

Not all limits provided by other runtimes are currently supported by SingularityCE. Specifically, the `--device-` flags supported by the `docker` CLI are not yet available.

CPU Limits

`--cpus` sets the number of CPUs, or fractional CPUs, that the container can use. The minimum is `0.01` or one hundredth of a physical CPU. The maximum is the number of CPU cores on your system.

```
# Limit container to 3.5 CPUs
$ singularity run --cpus 3.5 myfirstapp.sif
```

`--cpu-shares` sets a relative weight for a container's access to the system's CPUs, versus other containers that also have a `--cpus-shares` value set. If container A has 1024 cpu shares, and container B has 512 cpu shares, then container A will receive twice as much CPU time than container B, but *only when there is contention for CPUs*, i.e. the containers are able to consume more CPU time than is available.

```
# Container A - twice as much CPU priority as container B
$ singularity run --cpu-shares 1024 myfirstapp.sif

# Container A - half as much CPU priority as container A
$ singularity run --cpu-shares 512 mysecondapp.sif
```

`--cpu-set-cpus` specifies a list of physical CPU IDs on which a container can run. For example, on a dual CPU system you might pin one container to the first 12 cores on CPU 1, and a second container to the second 12 cores on CPU 2.

`--cpu-set-mems` specifies a list of memory nodes the container can use. It should generally be set to the same value as `--cpu-set-cpus`.

```
# Container A - first CPU
$ singularity run --cpu-set-cpus 0-11 --cpu-set-mems 0-11 myfirstapp.sif

# Container B - second CPU
$ singularity run --cpu-set-cpus 12-23 --cpu-set-mems 12-23 myfirstapp.sif
```

Memory Limits

`--memory` sets the maximum amount of RAM that a container can use, in bytes. You can use suffixes such as `M` or `G` to specify megabytes or gigabytes. If the container tries to use more memory than its limit, the system will kill it.

```
# Run a program that will use 10GB of RAM, with a 100MB limit
$ singularity exec --memory 100M memhog.sif memhog 10G
...............................................Killed
```

`--memory-reservation` sets a soft limit, which should be lower than the hard limit set with `--memory`. When there is contention for memory, the system will attempt to make sure the container doesn't exceed the soft limit.

```
# Kill my program if it exceeds 10G, but aim for 8G if there is contention
$ singularity exec --memory 10G --memory-reservation 8G myfirstapp.sif
```

`--memory-swap` sets the total amount of memory and swap space that a container can use. You must set `--memory` along with `--memory-swap`. A value of `-1` means *unlimited swap*. If `--memory-swap` is not set or is `0`, then the container can use an amount of swap up to the value of `--memory`. It's easier to understand this flag with examples:

```
# Run a container that can use up to 1G RAM, or swap if it is swapped out
$ singularity run --memory 1G myfirstapp.sif

# Run a container that can use up to 1G RAM, and no swap space
$ singularity run --memory 1G --memory-swap 1G myfirstapp.sif

# Run a container that can use up to 1G RAM, and unlimited swap space
$ singularity run --memory 1G --memory-swap -1 myfirstapp.sif

# Run a container that can use up to 1G RAM, and 1G swap space
$ singularity run --memory 1G --memory-swap 2G myfirstapp.sif
```

IO Limits

Note: Requires the `cfq` or `bfq` IO scheduler to be configured for block IO on the system. This is common on modern distributions, but not universal. Ask your system administrator if IO limits are not working as expected.

`--blkio-weight` sets a relative weight for the container when performing block I/O, e.g. reading/writing to/from disk. The weight should be between 10 and 1000, and will control how much I/O access a container recieves when there is contention for I/O with other containers. It may be useful to give high priority to a container that needs infrequent but time sensitive data access, running alongside an application that is continuously performing bulk reads.

```
# Container A - ten times as much block IO priority as container B
$ singularity run --blkio-weight 1000 myfirstapp.sif

# Container A - ten times less block IO priority as container A
$ singularity run --blkio-weight 100 mysecondapp.sif
```

`--blkio-weight-device` sets a relative weight for the container when performing block I/O on a specific device. Specify the device and weight as `<device path>:weight`:

```
# Container A - ten times as much block IO priority as container B on disk /dev/sda
$ singularity run --blkio-weight-device /dev/sda:1000 myfirstapp.sif

# Container A - ten times less block IO priority as container A on disk /dev/sda
$ singularity run --blkio-weight-device /dev/sda:100 mysecondapp.sif
```

5.8.3 Applying Resource Limits From a TOML file

SingularityCE 3.9 and above can directly apply resource limitations to systems configured for both cgroups v1 and the v2 unified hierarchy, using the `--apply-cgroups` flag. Resource limits are specified using a TOML file that represents the *resources* section of the OCI runtime-spec: https://github.com/opencontainers/runtime-spec/blob/master/config-linux.md#control-groups

On a cgroups v1 system the resources configuration is applied directly. On a cgroups v2 system the configuration is translated and applied to the unified hierarchy.

Under cgroups v1, access restrictions for device nodes are managed directly. Under cgroups v2, the restrictions are applied by attaching eBPF programs that implement the requested access controls.

To apply resource limits to a container, using the `--apply-cgroups` flag, which takes a path to a TOML file specifying the cgroups configuration to be applied:

```
$ singularity shell --apply-cgroups /path/to/cgroups.toml my_container.sif
```

Note: Using `--apply-cgroups` as a non-root user requires a cgroups v2 system, configured to use the `systemd cgroups` manager in `singularity.conf`.

CPU Limits

CPU usage can be limited using different strategies, with limits specified in the `[cpu]` section of the TOML file.

shares

This corresponds to a ratio versus other cgroups with cpu shares. Usually the default value is `1024`. That means if you want to allow to use 50% of a single CPU, you will set `512` as value.

```
[cpu]
    shares = 512
```

A cgroup can get more than its share of CPU if there are enough idle CPU cycles available in the system, due to the work conserving nature of the scheduler, so a contained process can consume all CPU cycles even with a ratio of 50%. The ratio is only applied when two or more processes conflicts with their needs of CPU cycles.

quota/period

You can enforce hard limits on the CPU cycles a cgroup can consume, so contained processes can't use more than the amount of CPU time set for the cgroup. `quota` allows you to configure the amount of CPU time that a cgroup can use per period. The default is 100ms (100000us). So if you want to limit amount of CPU time to 20ms during period of 100ms:

```
[cpu]
    period = 100000
    quota = 20000
```

cpus/mems

You can also restrict access to specific CPUs (cores) and associated memory nodes by using `cpus/mems` fields:

```
[cpu]
    cpus = "0-1"
    mems = "0-1"
```

Where the container has limited access to CPU 0 and CPU 1.

Note: It's important to set identical values for both `cpus` and `mems`.

Memory Limits

To limit the amount of memory that your container uses to 500MB (524288000 bytes), set a `limit` value inside the `[memory]` section of your cgroups TOML file:

```
[memory]
    limit = 524288000
```

Start your container, applying the toml file, e.g.:

```
$ singularity run --apply-cgroups path/to/cgroups.toml library://alpine
```

After that, you can verify that the container is only using 500MB of memory. This example assumes that there is only one running container. If you are running multiple containers you will find multiple cgroups trees under the `singularity` directory.

```
# cgroups v1
$ cat /sys/fs/cgroup/memory/singularity/*/memory.limit_in_bytes
  524288000

# cgroups v2 - note translation of memory.limit_in_bytes -> memory.max
$ cat /sys/fs/cgroup/singularity/*/memory.max
524288000
```

IO Limits

To control block device I/O, applying limits to competing container, use the `[blockIO]` section of the TOML file:

```
[blockIO]
    weight = 1000
    leafWeight = 1000
```

`weight` and `leafWeight` accept values between `10` and `1000`.

`weight` is the default weight of the group on all the devices until and unless overridden by a per device rule.

`leafWeight` relates to weight for the purpose of deciding how heavily to weigh tasks in the given cgroup while competing with the cgroup's child cgroups.

To apply limits to specific block devices, you must set configuration for specific device major/minor numbers. For example, to override `weight/leafWeight` for `/dev/loop0` and `/dev/loop1` block devices, set limits for device major 7, minor 0 and 1:

```
[blockIO]
    [[blockIO.weightDevice]]
        major = 7
        minor = 0
        weight = 100
```

(continues on next page)

(continued from previous page)

```
        leafWeight = 50
    [[blockIO.weightDevice]]
        major = 7
        minor = 1
        weight = 100
        leafWeight = 50
```

You can also limit the IO read/write rate to a specific absolute value, e.g. 16MB per second for the `/dev/loop0` block device. The `rate` is specified in bytes per second.

```
[blockIO]
    [[blockIO.throttleReadBpsDevice]]
        major = 7
        minor = 0
        rate = 16777216
    [[blockIO.throttleWriteBpsDevice]]
        major = 7
        minor = 0
        rate = 16777216
```

Device Limits

Note: Device limits can only be applied when running as the root user, and will be ignored as a non-root user.

You can limit read (`r`), write (`w`), or creation (`c`) of devices by a container. Like applying I/O limits to devices, you must use device node major and minor numbers to create rules for specific devices or classes of device.

In this example, a container is configured to only be able to read from or write to `/dev/null`:

```
[[devices]]
    access = "rwm"
    allow = false
[[devices]]
    access = "rw"
    allow = true
    major = 1
    minor = 3
    type = "c"
```

Other limits

SingularityCE can apply all resource limits that are valid in the OCI runtime-spec `resources` section. If you use cgroups v1 limits on a cgroups v2 system they will be translated at runtime. You may also specify native cgroups v2 limits under the `unified` key.

See https://github.com/opencontainers/runtime-spec/blob/master/config-linux.md#control-groups for information about the available limits. Note that SingularityCE uses TOML format for the configuration file, rather than JSON.

5.8.4 Applying Resource Limits With External Tools

Because SingularityCE starts a container as a simple process, rather than using a daemon, you can limit resource usage by running the `singularity` command inside an existing cgroup. This is convenient where, for example, a job scheduler uses cgroups to control job limits. By running `singularity` inside your batch script, your container will respect the limits set by the scheduler on the job's cgroup.

systemd-run

As well as schedulers you can use tools such as `systemd-run` to create a cgroup, and run SingularityCE inside of it. This is convenient on modern cgroups v2 systems, where the creation of cgroups can be delegated to users through systemd. Without this delegation `root` privileges are required to create a cgroup.

For example, assuming your system is configured correctly for unprivileged cgroup creation via systemd, you can limit the number of CPUs a container run is allowed to use:

```
$ systemd-run --user --scope -p AllowedCPUs=1,2 -- singularity run mycontainer.sif
```

- `--user` instructs systemd that we want to run as our own user account.
- `--scope` will run our command in an interactive scope that inherits from our environment. By default the command would run as a service, in the background.
- `-p AllowedCPUs=1,2` sets a property on our scope, so that in this case systemd will then setup a cgroup limiting our command to using CPU 1 and 2 only.
- The double hyphen `--` separates options for `systemd-run` from the actual command we wish to execute. This is important so that `systemd-run` doesn't capture any flags we might need to pass to `singularity`.

You can read more about how systemd can control resources uses at the link below, which details the properties you can set using `systemd-run`.

https://www.freedesktop.org/software/systemd/man/systemd.resource-control.html

CHAPTER

SIX

COMPATIBILITY

SingularityCE has unique benefits and supports easy access to GPUs and other hardware. It also strives for compatibility with Docker/OCI container formats. Understand the differences between SingularityCE and Docker, as well as how to use containerized MPI and GPU applications.

6.1 Support for Docker and OCI Containers

The Open Containers Initiative (OCI) container format, which grew out of Docker, is the dominant standard for cloud-focused containerized deployments of software. Although SingularityCE's own container format has many unique advantages, it's likely you will need to work with Docker/OCI containers at some point.

SingularityCE's default native mode aims for maximum compatibility with Docker, within the constraints on a runtime that is well suited for use on shared systems and especially in HPC environments that often employ older LTS Linux distributions.

SingularityCE's optional OCI-mode (`--oci`) uses a true OCI low-level runtime, and encapsulates OCI images into an OCI-SIF file, rather than converting them to SingularityCE's own container format. It offers improved OCI compatibility but has additional requirements. Please *consult the documentation* about OCI-mode for more information.

Using SingularityCE you can:

- Pull, run, and build from most containers on Docker Hub, without changes.
- Pull, run, and build from containers hosted on other registries, including private registries deployed on premise, or in the cloud.
- Pull and build from OCI containers in archive formats, or cached in a local Docker daemon.

This section will highlight these workflows, and discuss the limitations and best practices to keep in mind when creating containers targeting both Docker and SingularityCE.

6.1.1 Containers From Docker Hub

Docker Hub is the most common place that projects publish public container images. At some point, it's likely that you will want to run or build from containers that are hosted there.

Public Containers

It's easy to run a public Docker Hub container with SingularityCE. Just put `docker://` in front of the container repository and tag. To run the container that's called `sylabsio/lolcow:latest`:

```
$ singularity run docker://sylabsio/lolcow:latest
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
Getting image source signatures
Copying blob 16ec32c2132b done
Copying blob 5ca731fc36c2 done
Copying config fd0daa4d89 done
Writing manifest to image destination
Storing signatures
2021/10/04 14:50:21  info unpack layer:␣
↪sha256:16ec32c2132b43494832a05f2b02f7a822479f8250c173d0ab27b3de78b2f058
2021/10/04 14:50:23  info unpack layer:␣
↪sha256:5ca731fc36c28789c5ddc3216563e8bfca2ab3ea10347e07554ebba1c953242e
INFO:    Creating SIF file...
 _______________________________
< Mon Oct 4 14:50:30 CDT 2021 >
 -------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

Note that SingularityCE retrieves blobs and configuration data from Docker Hub, extracts the layers that make up the Docker container, and creates a SIF file from them. This SIF file is kept in your SingularityCE *cache directory*, so if you run the same Docker container again the downloads and conversion aren't required.

To obtain the Docker container as a SIF file in a specific location, which you can move, share, and keep for later, `singularity pull` it:

```
$ singularity pull docker://sylabsio/lolcow
INFO:    Using cached SIF image

$ ls -l lolcow_latest.sif
-rwxr-xr-x 1 myuser myuser 74993664 Oct  4 14:55 lolcow_latest.sif
```

If it's the first time you pull the container it'll be downloaded and translated. If you have pulled the container before, it will be copied from the cache.

Note: `singularity pull` of a Docker container actually runs a `singularity build` behind the scenes, since we are translating from OCI to SIF. If you `singularity pull` a Docker container twice, the output file isn't identical because metadata such as dates from the conversion will vary. This differs from pulling a SIF container (e.g. from a `library://` URI), which always give you an exact copy of the image.

OCI-Mode

To use SingularityCE 4's new OCI-mode, add the `--oci` option when you `run / shell / exec` or `pull` an OCI container:

```
$ singularity run --oci docker://sylabsio/lolcow:latest
INFO:    Converting OCI image to OCI-SIF format
INFO:    Squashing image to single layer
INFO:    Writing OCI-SIF image
INFO:    Cleaning up.
 _____________________________
< Tue Sep 5 10:36:58 UTC 2023 >
 -----------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
```

Note that in this case, the log messages show that SingularityCE is converting the image to OCI-SIF format. This is closer to the original OCI image than a SIF created in native (non-OCI) mode. You can read more in the *OCI-SIF section* of this documentation.

When you `pull` an image with `--oci`, the OCI-SIF file will have an `.oci.sif` extension by default:

```
$ singularity pull --oci docker://sylabsio/lolcow
INFO:    Using cached OCI-SIF image

$ ls -l lolcow_latest.oci.sif
-rwxr-xr-x. 1 myuser myuser 74728057 Sep  5 11:39 lolcow_latest.oci.sif
```

Docker Hub Limits

Docker Hub introduced limits on anonymous access to its API in November 2020. Every time you use a `docker://` URI to run, pull etc. a container SingularityCE will make requests to Docker Hub in order to check whether the container has been modified there. On shared systems, and when running containers in parallel, this can quickly exhaust the Docker Hub API limits.

We recommend that you `singularity pull` a Docker image to a local SIF, and then always run from the SIF file, rather than using `singularity run docker://...` repeatedly.

Alternatively, if you have signed up for a Docker Hub account, make sure that you authenticate before using `docker://` container URIs.

Authentication / Private Containers

To make use of the API limits under a Docker Hub account, or to access private containers, you'll need to authenticate to Docker Hub. There are a number of ways to do this with SingularityCE.

Singularity CLI `registry` Command

The `singularity registry login` command supports logging into Docker Hub and other OCI registries. For Docker Hub, the registry hostname is `docker.io`, so you will need to login as below, specifying your username:

```
$ singularity registry login --username myuser docker://docker.io
Password / Token:
INFO:    Token stored in /home/myuser/.singularity/remote.yaml
```

The Password / Token you enter must be a Docker Hub CLI access token, which you should generate in the 'Security' section of your account profile page on Docker Hub.

To check which Docker / OCI registries you are currently logged in to, use `singularity registry list`.

To logout of a registry, so that your credentials are forgotten, use `singularity registry logout`:

```
$ singularity registry logout docker://docker.io
INFO:    Logout succeeded
```

Docker CLI Authentication

If you have the `docker` CLI installed on your machine, you can `docker login` to your account. This stores authentication information in `~/.docker/config.json`. The process that SingularityCE uses to retrieve Docker / OCI containers will attempt to use this information to login.

Note: SingularityCE can only read credentials stored directly in `~/.docker/config.json`. It cannot read credentials from external Docker credential helpers.

Interactive Login

To perform a one-off interactive login, which will not store your credentials, use the `--docker-login` flag:

```
$ singularity pull --docker-login docker://sylabsio/private
Enter Docker Username: myuser
Enter Docker Password:
```

Environment Variables

When calling SingularityCE in a CI/CD workflow, or other non-interactive scenario, it may be useful to specify Docker Hub login credentials using environment variables. These are often the default way of passing secrets into jobs within CI pipelines.

Singularity accepts a username, and password / token, as `SINGULARITY_DOCKER_USERNAME` and `SINGULARITY_DOCKER_PASSWORD` respectively. These environment variables will override any stored credentials.

If `DOCKER_USERNAME` and `DOCKER_PASSWORD`, without the `SINGULARITY_` prefix, are set they will also be used provided the `SINGULARITY_` equivalent is not overriding them. This allows a single set of environment variables to be set for both `singularity` and `docker` operations.

```
$ export SINGULARITY_DOCKER_USERNAME=myuser
$ export SINGULARITY_DOCKER_PASSWORD=mytoken
$ singularity pull docker://sylabsio/private
```

6.1.2 Containers From Other Registries

You can use `docker://` URIs with SingularityCE to pull and run containers from OCI registries other than Docker Hub. To do this, you'll need to include the hostname or IP address of the registry in your `docker://` URI. Authentication with other registries is carried out in the same basic manner, but sometimes you'll need to retrieve your credentials using a specific tool, especially when working with Cloud Service Provider environments.

Below are specific examples for some common registries. Most other registries follow a similar pattern for pulling public images, and authenticating to access private images.

Quay.io

Quay is an OCI container registry used by a large number of projects, and hosted at `https://quay.io`. To pull public containers from Quay, just include the `quay.io` hostname in your `docker://` URI:

```
$ singularity pull docker://quay.io/bitnami/python:3.7
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
...

$ singularity run python_3.7.sif
Python 3.7.12 (default, Sep 24 2021, 11:48:27)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

To pull containers from private repositories you will need to generate a CLI token in the Quay web interface, then use it to login with SingularityCE. Use the same methods as described for Docker Hub above:

- Run `singularity registry login --username myuser docker://quay.io` to store your credentials for SingularityCE.
- Use `docker login quay.io` if `docker` is on your machine.
- Use the `--docker-login` flag for a one-time interactive login.
- Set the `SINGULARITY_DOCKER_USERNAME` and `SINGULARITY_DOCKER_PASSWORD` environment variables.

NVIDIA NGC

The NVIDIA NGC catalog at https://ngc.nvidia.com contains various GPU software, packaged in containers. Many of these containers are specifically documented by NVIDIA as supported by SingularityCE, with instructions available.

Previously, an account and API token was required to pull NGC containers. However, they are now available to pull as a guest without login:

```
$ singularity pull docker://nvcr.io/nvidia/pytorch:21.09-py3
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
```

If you do need to pull containers using an NVIDIA account, e.g. if you have access to an NGC Private Registry, you will need to generate an API key in the web interface in order to authenticate.

Use one of the following authentication methods (detailed above for Docker Hub), with the username `$oauthtoken` and the password set to your NGC API key.

- Run `singularity registry login --username \$oauthtoken docker://nvcr.io` to store your credentials for SingularityCE.
- Use `docker login nvcr.io` if `docker` is on your machine.
- Use the `--docker-login` flag for a one-time interactive login.
- Set the `SINGULARITY_DOCKER_USERNAME="\$oauthtoken"` and `SINGULARITY_DOCKER_PASSWORD` environment variables.

See also: https://docs.nvidia.com/ngc/ngc-private-registry-user-guide/index.html

GitHub Container Registry

GitHub Container Registry is increasingly used to provide Docker containers alongside the source code of hosted projects. You can pull a public container from GitHub Container Registry using a `ghcr.io` URI:

```
$ singularity pull docker://ghcr.io/containerd/alpine:latest
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
```

To pull private containers from GHCR you will need to generate a personal access token in the GitHub web interface in order to authenticate. This token must have required scopes. See the GitHub documentation here.

Use one of the following authentication methods (detailed above for Docker Hub), with your username and personal access token:

- Run `singularity registry login --username myuser docker://ghcr.io` to store your credentials for SingularityCE.
- Use `docker login ghcr.io` if `docker` is on your machine.
- Use the `--docker-login` flag for a one-time interactive login.
- Set the `SINGULARITY_DOCKER_USERNAME` and `SINGULARITY_DOCKER_PASSWORD` environment variables.

AWS ECR

To work with an AWS hosted Elastic Container Registry (ECR) generally requires authentication. There are various ways to generate credentials. You should follow one of the approaches in the ECR guide in order to obtain a username and password.

> **Warning:** The ECR Docker credential helper cannot be used, as SingularityCE does not currently support external credential helpers used with Docker, only reading credentials stored directly in the `.docker/config.json` file.

The `get-login-password` approach is the most straightforward. It uses the AWS CLI to request a password, which can then be used to authenticate to an ECR private registry in the specified region. The username used in conjunction with this password is always `AWS`.

```
$ aws ecr get-login-password --region region
```

Then login using one of the following methods:

- Run `singularity registry login --username AWS docker://<accountid>.dkr.ecr.<region>.amazonaws.com` to store your credentials for SingularityCE.
- Use `docker login --username AWS <accountid>.dkr.ecr.<region>.amazonaws.com` if `docker` is on your machine.
- Use the `--docker-login` flag for a one-time interactive login.
- Set the `SINGULARITY_DOCKER_USERNAME=AWS` and `SINGULARITY_DOCKER_PASSWORD` environment variables.

You should now be able to pull containers from your ECR URI at `docker://<accountid>.dkr.ecr.<region>.amazonaws.com`.

Azure ACR

An Azure hosted Azure Container Registry (ACR) will generally hold private images and require authentication to pull from. There are several ways to authenticate to ACR, depending on the account type you use in Azure. See the ACR documentation for more information on these options.

Generally, for identities, using `az acr login` from the Azure CLI will add credentials to `.docker/config.json` which can be read by SingularityCE.

Service Principle accounts will have an explicit username and password, and you should authenticate using one of the following methods:

- Run `singularity registry login --username myuser docker://myregistry.azurecr.io` to store your credentials for SingularityCE.
- Use `docker login --username myuser myregistry.azurecr.io` if `docker` is on your machine.
- Use the `--docker-login` flag for a one-time interactive login.
- Set the `SINGULARITY_DOCKER_USERNAME` and `SINGULARITY_DOCKER_PASSWORD` environment variables.

The recent repository-scoped access token preview may be more convenient. See the preview documentation which details how to use `az acr token create` to obtain a token name and password pair that can be used to authenticate with the above methods.

6.1.3 Specifying a platform / architecture

By default, `singularity pull` from a `docker://` URI will attempt to fetch a container that matches the architecture of your host system. If you need to retrieve a container that does not have the same architecture as your host (e.g. an `arm64` container on an `amd64` host), you can use the `--platform` or `--arch` options.

`--platform` option

The `--platform` option for `singularity pull` accepts an OCI platform string. This has two or three parts, separated by forward slashes (`/`):

- An OS value. Only `linux` is supported by SingularityCE.
- A CPU architecture value, e.g. `arm64`.
- An optional CPU variant, e.g. `v8`.

For example, the platform string for a 32-bit v7 ARM container would be `linux/arm/v7`.

SingularityCE will normalize any platform string you supply, before passing it to the OCI registry, to ensure that it matches intended images.

To pull an Ubuntu image for a 64-bit ARM system from Docker Hub, using the `--platform` option:

```
$ singularity pull --platform linux/arm64 docker://ubuntu
```

To pull a 32-bit image for a v7 ARM CPU:

```
$ singularity pull --platform linux/arm/v7 docker://ubuntu
```

`--arch` option

The `--arch` option accepts a CPU architecture only. For example, to pull an Ubuntu image for a 64-bit ARM system:

```
$ singularity pull --arch arm64 docker://ubuntu
```

CPU emulation

If you try to run a container that does not match the host CPU architecture, it will likely fail:

```
$ singularity run ppc64le.sif
FATAL:   While checking image: could not open image /home/dtrudg-sylabs/Git_Sylabs/
↪singularity-userdocs/ppc64le.sif: the image's architecture (ppc64le) could not run on␣
↪the host's (amd64)
```

However, SingularityCE is able to make use of CPU emulation with QEMU, and the Linux kernel's binfmt_misc mechanism, to run containers that do not match the host CPU.

An adminstrator can configure emulation support by installing distribution packages, or using the multiarch/qemu-user-static container from Docker Hub:

```
$ sudo singularity run docker://multiarch/qemu-user-static --reset -p yes
```

Note: Running this container with sudo will modify system configuration files, and register binaries on the host.

It is now possible to run containers for other architectures:

```
# The host system is an AMD64 / x86_64 machine
$ uname -m
x86_64

# A ppc64le container can be run using emulation
$ singularity run ppc64le.sif uname -m
ppc64le
```

Running a container in this manner, using emulation, will be many times slower than running on a system where the CPU architecture matches the container. Emulation is often useful for testing and development purposes, but rarely appropriate when deploying a container to an HPC system.

6.1.4 Building From Docker / OCI Containers

If you wish to use an existing Docker or OCI container as the basis for a new container, you will need to specify it as the *bootstrap* source in a SingularityCE definition file.

Just as you can run or pull containers from different registries using a `docker://` URI, you can use different headers in a definition file to instruct SingularityCE where to find the container you want to use as the starting point for your build.

Note: OCI-mode doesn't yet support `singularity build`. When you build from an OCI container with SingularityCE, you are always creating a non-OCI SingularityCE container as output.

Registries In Definition Files

When you wish to build from a Docker or OCI container that's hosted in a registry, such as Docker Hub, your definition file should begin with `Bootstrap: docker`, followed with a `From:` line which specifies the location of the container you wish to pull.

Docker Hub

Docker Hub is the default registry, so when building from Docker Hub the `From:` header only needs to specify the container repository and tag:

```
Bootstrap: docker
From: ubuntu:20.04
```

If you `singularity build` a definition file with these lines, SingularityCE will fetch the `ubuntu:20.04` container image from Docker Hub, and extract it as the basis for your new container.

Other Registries

To pull from a different Docker registry, you can either specify the hostname in the `From:` header, or use the separate `Registry:` header. The following two examples are equivalent:

```
Bootstrap: docker
From: quay.io/bitnami/python:3.7
```

```
Bootstrap: docker
Registry: quay.io
From: bitnami/python:3.7
```

Authentication During a Build

If you are building from an image in a private registry you will need to ensure that the credentials needed to access the image are available to SingularityCE.

A build might be run as the `root` user, e.g. via `sudo`, or under your own account with `--fakeroot`.

If you are running the build as `root`, using `sudo`, then any stored credentials or environment variables must be available to the `root` user:

- Use the `--docker-login` flag for a one-time interactive login. I.E. run `sudo singularity build --docker-login myimage.sif Singularity`.
- Set the `SINGULARITY_DOCKER_USERNAME` and `SINGULARITY_DOCKER_PASSWORD` environment variables. Pass the environment variables through sudo to the `root` build process by running `sudo -E singularity build ...`.
- Run `sudo singularity registry login ...` to store your credentials for the `root` user on your system. This is separate from storing the credentials under your own account.
- Use `sudo docker login` if `docker` is on your machine. This is separate from storing the credentials under your own account.

If you are running the build under your account via the `--fakeroot` feature you do not need to specially set credentials for the root user.

Archives & Docker Daemon

As well as being hosted in a registry, Docker / OCI containers might be found inside a running Docker daemon, or saved as an archive. SingularityCE can build from these locations by using specialized bootstrap agents.

Containers Cached by the Docker Daemon

If you have pulled or run a container on your machine under `docker`, it will be cached locally by the Docker daemon. The `docker images` command will list containers that are available:

```
$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
sylabsio/lolcow     latest              5a15b484bc65        2 hours ago         188MB
```

This indicates that `sylabsio/lolcow:latest` has been cached locally by Docker. You can directly build it into a SIF file using a `docker-daemon:` URI specifying the `REPOSITORY:TAG` container name:

```
$ singularity build lolcow_from_docker_cache.sif docker-daemon:sylabsio/lolcow:latest
INFO:    Starting build...
Getting image source signatures
Copying blob sha256:a2022691bf950a72f9d2d84d557183cb9eee07c065a76485f1695784855c5193
 119.83 MiB / 119.83 MiB [==========================================================] 6s
Copying blob sha256:ae620432889d2553535199dbdd8ba5a264ce85fcdcd5a430974d81fc27c02b45
 15.50 KiB / 15.50 KiB [============================================================] 0s
Copying blob sha256:c561538251751e3685c7c6e7479d488745455ad7f84e842019dcb452c7b6fecc
 14.50 KiB / 14.50 KiB [============================================================] 0s
Copying blob sha256:f96e6b25195f1b36ad02598b5d4381e41997c93ce6170cab1b81d9c68c514db0
 5.50 KiB / 5.50 KiB [==============================================================] 0s
Copying blob sha256:7f7a065d245a6501a782bf674f4d7e9d0a62fa6bd212edbf1f17bad0d5cd0bfc
```

(continues on next page)

(continued from previous page)

```
 3.00 KiB / 3.00 KiB [=======================================================] 0s
Copying blob sha256:70ca7d49f8e9c44705431e3dade0636a2156300ae646ff4f09c904c138728839
 116.56 MiB / 116.56 MiB [=====================================================] 6s
Copying config sha256:73d5b1025fbfa138f2cacf45bbf3f61f7de891559fa25b28ab365c7d9c3cbd82
 3.33 KiB / 3.33 KiB [=======================================================] 0s
Writing manifest to image destination
Storing signatures
INFO:    Creating SIF file...
INFO:    Build complete: lolcow_from_docker_cache.sif
```

The tag name must be included in the URI. Unlike when pulling from a registry, the `docker-daemon` bootstrap agent will not try to pull a `latest` tag automatically.

Note: In the example above, the build was performed without `sudo`. This is possible only when the user is part of the `docker` group on the host, since SingularityCE must contact the Docker daemon through its socket. If you are not part of the `docker` group you will need to use `sudo` for the build to complete successfully.

To build from an image cached by the Docker daemon in a definition file use `Bootstrap: docker-daemon`, and a `From: <REPOSITORY>:TAG` line:

```
Bootstrap: docker-daemon
From: sylabsio/lolcow:latest
```

Containers in Docker Archive Files

Docker allows containers to be exported into single file tar archives. These cannot be run directly, but are intended to be imported into Docker to run at a later date, or another location. SingularityCE can build from (or run) these archive files, by extracting them as part of the build process.

If an image is listed by the `docker images` command, then we can create a tar archive file using `docker save` and the image ID:

```
$ sudo docker images
REPOSITORY                      TAG                 IMAGE ID       CREATED          SIZE
sylabsio/lolcow                 latest              5a15b484bc65   2 hours ago      188MB

$ docker save 5a15b484bc65 -o lolcow.tar
```

If we examine the contents of the tar file we can see that it contains the layers and metadata that make up a Docker container:

```
$ tar tvf lolcow.tar
drwxr-xr-x  0 0      0           0 Aug 16 11:22␣
↪2f0514a4c044af1ff4f47a46e14b6d46143044522fcd7a9901124209d16d6171/
-rw-r--r--  0 0      0           3 Aug 16 11:22␣
↪2f0514a4c044af1ff4f47a46e14b6d46143044522fcd7a9901124209d16d6171/VERSION
-rw-r--r--  0 0      0         401 Aug 16 11:22␣
↪2f0514a4c044af1ff4f47a46e14b6d46143044522fcd7a9901124209d16d6171/json
-rw-r--r--  0 0      0    75156480 Aug 16 11:22␣
↪2f0514a4c044af1ff4f47a46e14b6d46143044522fcd7a9901124209d16d6171/layer.tar
-rw-r--r--  0 0      0        1499 Aug 16 11:22␣
```

(continues on next page)

(continued from previous page)

```
↪5a15b484bc657d2b418f2c20628c29945ec19f1a0c019d004eaf0ca1db9f952b.json
drwxr-xr-x  0 0      0           0 Aug 16 11:22␣
↪af7e389ea6636873dbc5adc17826e8401d96d3d384135b2f9fe990865af202ab/
-rw-r--r--  0 0      0           3 Aug 16 11:22␣
↪af7e389ea6636873dbc5adc17826e8401d96d3d384135b2f9fe990865af202ab/VERSION
-rw-r--r--  0 0      0         946 Aug 16 11:22␣
↪af7e389ea6636873dbc5adc17826e8401d96d3d384135b2f9fe990865af202ab/json
-rw-r--r--  0 0      0   118356480 Aug 16 11:22␣
↪af7e389ea6636873dbc5adc17826e8401d96d3d384135b2f9fe990865af202ab/layer.tar
-rw-r--r--  0 0      0         266 Dec 31  1969 manifest.json
```

We can convert this tar file into a singularity container using the `docker-archive` bootstrap agent. Because the agent accesses a file, rather than an object hosted by a service, it uses `:<filename>`, not `://<location>`. To build a tar archive directly to a SIF container:

```
$ singularity build lolcow_tar.sif docker-archive:lolcow.tar
INFO:    Starting build...
Getting image source signatures
Copying blob sha256:2f0514a4c044af1ff4f47a46e14b6d46143044522fcd7a9901124209d16d6171
 119.83 MiB / 119.83 MiB [=========================================================] 6s
Copying blob sha256:af7e389ea6636873dbc5adc17826e8401d96d3d384135b2f9fe990865af202ab
 15.50 KiB / 15.50 KiB [===========================================================] 0s
Copying config sha256:5a15b484bc657d2b418f2c20628c29945ec19f1a0c019d004eaf0ca1db9f952b
 3.33 KiB / 3.33 KiB [=============================================================] 0s
Writing manifest to image destination
Storing signatures
INFO:    Creating SIF file...
INFO:    Build complete: lolcow_tar.sif
```

Note: The `docker-archive` bootstrap agent can also handle gzipped Docker archives (`.tar.gz` or `.tgz` files).

To build an image using a definition file, which starts from a container in a Docker archive, use `Bootstrap: docker-archive` and specify the filename in the `From:` line:

```
Bootstrap: docker-archive
From: lolcow.tar
```

6.1.5 Differences and Limitations vs Docker

Though Docker / OCI container compatibility is a goal of SingularityCE, there are some differences and limitations due to the way SingularityCE was designed to work well on shared systems and HPC clusters, particularly for the native (non-OCI) mode.

If you are having difficulty running a specific Docker container, without `--oci`, check through the list of differences below. There are workarounds for many of the issues that you are most likely to face. You may also wish to use OCI-mode for improved compatibility.

Read-only by Default

SingularityCE's container image format (SIF) is generally read-only. This permits containers to be run in parallel from a shared location on a network filesystem, support in-built signing and verification, and offer encryption. A container's filesystem is mounted directly from the SIF, as SquashFS, so cannot be written to by default.

When a container is run using Docker its layers are extracted, and the resulting container filesystem can be written to and modified by default. If a Docker container expects to write files, you will need to follow one of the following methods to allow it to run under SingularityCE.

- A directory from the host can be passed into the container with the `--bind` or `--mount` flags. It needs to be mounted inside the container at the location where files will be written.
- The `--writable-tmpfs` flag can be used to allow files to be created in a special temporary overlay. Any changes are lost when the container exits. The SIF file is never modified.
- The container can be converted to a sandbox directory, and executed with the `--writable` flag, which allows modification of the sandbox content.
- A writable overlay partition can be added to the SIF file, and the container executed with the `--writable` flag. Any changes made are kept permanently in the overlay partition.

Of these methods, only `--writable-tmpfs` is always safe to run in parallel. Each time the container is executed, a separate temporary overlay is used and then discarded.

Binding a directory into a container, or running a writable sandbox may or may not be safe, depending on the program executed. The program must use, and the filesystem support, some type of locking in order that the parallel runs do not interfere.

A writable overlay file in a SIF partition cannot be used in parallel. SingularityCE will refuse to run concurrently using the same SIF writable overlay partition.

Note: Using *–writable-tmpfs* as a non-root user requires that SingularityCE was installed in setuid mode, or the system has a kernel version >=5.11 in non-setuid mode.

Using a writable overlay as a non-root user generally requires that SingularityCE was installed in setuid mode.

The `--writable-tmpfs` size is controlled by `sessiondir max size` in `singularity.conf`. This defaults to 64MiB, and may need to be increased if your workflows create larger temporary files.

Dockerfile `USER`

The `Dockerfile` used to build a Docker container may contain a `USER` statement. This tells the container runtime that it should run the container under the specified user account.

Because SingularityCE was designed to provide easy and safe access to data on the host system, in a manner that supports older Linux distributions, it does not permit changing the user account the container is run as.

In the default native mode, any `USER` statement in a `Dockerfile` will be ignored by SingularityCE when the container is run. In practice, this often does not affect the execution of the software in the container. Software that is written in a way that requires execution under a specific user account will generally require modification for use with SingularityCE.

Note: The new OCI-mode (`--oci`) supports running containers with the `USER` requested in a `Dockerfile`. It uses newer kernel features to achieve this. You may wish to use OCI-mode if your system supports it.

SingularityCE's `--fakeroot` mode will start a container as a fake `root` user, mapped to the user's real account outside of the container. Inside the container it is possible to change to another user account, which is mapped to a configured

range of sub-uids / gids belonging to the original user. It may be possible to execute software expecting a fixed user account manually inside a `--fakeroot` shell, if your adminstrator has configured the system for `--fakeroot`.

Default Mounts / $HOME

A default installation of SingularityCE will mount the user's home directory, `/tmp` directory, and the current working directory, into each container that is run. Administrators may also configure e.g. HPC project directories to automatically bind mount. Docker does not mount host directories into the container by default.

The home directory mount is the most likely to cause problems when running Docker containers. Various software will look for packages, plugins, and configuration files in `$HOME`. If you have, for example, installed packages for Python into your home directory (`pip install --user`) then a Python container may find and attempt to use them. This can cause conflicts and unexpected behavior.

If you experience issues, use the `--contain` option to stop SingularityCE automatically binding directories into the container. You may need to use `--bind` or `--mount` to then add back e.g. an HPC project directory that you need access to.

```
# Without --contain, python in the container finds packages
# in your $HOME directory.
$ singularity exec docker://python:3.9 pip list
Package    Version
---------- -------
pip        21.2.4
rstcheck   3.3.1
setuptools 57.5.0
wheel      0.37.0

# With --contain, python in the container only finds packages
# installed in the container.
$ singularity exec --contain docker://python:3.9 pip list
Package    Version
---------- -------
pip        21.2.4
setuptools 57.5.0
wheel      0.37.0
```

Environment Propagation

SingularityCE propagates most environment variables set on the host into the container, by default. Docker does not propagate any host environment variables into the container. Environment variables may change the behaviour of software.

To disable automatic propagation of environment variables, the `--cleanenv / -e` flag can be specified. When `--cleanenv` is used, only variables on the host that are prefixed with `SINGULARITYENV_` are set in the container:

```
# Set a host variable
$ export HOST_VAR=123
# Set a singularity container environment variable
$ export "SINGULARITYENV_FORCE_VAR="123"

$ singularity run library://alpine env | grep VAR
FORCE_VAR=123
```

(continues on next page)

(continued from previous page)

```
HOST_VAR=ABC

$ singularity run --cleanenv library://alpine env | grep VAR
FORCE_VAR=123
```

Any environment variables set via an `ENV` line in a `Dockerfile` will be available when the container is run with SingularityCE. You can override them with `SINGULARITYENV_` vars, or the `--env / --env-file` flags, but they will not be overridden by host environment variables.

For example, the `docker://openjdk:latest` container sets `JAVA_HOME`:

```
# Set a host JAVA_HOME
export JAVA_HOME=/test

# Check JAVA_HOME in the docker container.
# This value comes from ENV in the Dockerfile.
$ singularity run docker://openjdk:latest echo \$JAVA_HOME
/usr/java/openjdk-17

# Override JAVA_HOME in the container
export SINGULARITYENV_JAVA_HOME=/test
$ singularity run docker://openjdk:latest echo \$JAVA_HOME
/test
```

Environment Variable Escaping / Evaluation

The default behavior of SingularityCE, in native mode, differs from Docker/OCI handling of environment variables as SingularityCE uses a shell interpreter to process environment on container startup, in a manner that evaluates environment variables. To avoid the extra evaluation of variables that SingularityCE performs you can:

- Follow the instructions in the *Escaping and Evaluation of Environment Variables* section to explictly escape environment variables.
- Use the `--no-eval` flag, or `--compat` (which enables `--no-eval`).

Note: When running a container in OCI-mode (`--oci`), SingularityCE follows Docker/OCI behaviour by default. You do not need to enable the `--no-eval` or `--compat` options.

`--no-eval` prevents SingularityCE from evaluating environment variables on container startup, so that they will take the same value as with a Docker/OCI runtime:

```
# Set an environment variable that would run `date` if evaluated
$ export SINGULARITYENV_MYVAR='$(date)'

# Default behavior
# MYVAR was evaluated in the container, and is set to the output of `date`
$ singularity run ~/ubuntu_latest.sif env | grep MYVAR
MYVAR=Tue Apr 26 14:37:07 CDT 2022

# --no-eval / --compat behavior
# MYVAR was not evaluated and is a literal `$(date)`
```

(continues on next page)

(continued from previous page)

```
$ singularity run --no-eval ~/ubuntu_latest.sif env | grep MYVAR
MYVAR=$(date)
```

Namespace & Device Isolation

Because SingularityCE favors an integration over isolation approach it does not, by default, use all the methods through which a container can be isolated from the host system. This makes it much easier to run a SingularityCE container like any other program, while the unique security model ensures safety. You can access the host's network, GPUs, and other devices directly. Processes in the container are not numbered separately from host processes. Hostnames are not changed, etc.

Most containers are not impacted by the differences in isolation. If you require more isolation, than SingularityCE provides by default, you can enable some of the extra namespaces that Docker uses, with flags:

- `--ipc / -i` creates a separate IPC (inter process communication) namespace, for SystemV IPC objects and POSIX message queues.
- `--net / -n` creates a new network namespace, abstracting the container networking from the host.
- `--userns / -u` runs the container unprivileged, inside a user namespace and avoiding SingularityCE's setuid setup code. By default, SIF container images will be extracted to disk, as mounting the container filesystem from the SIF requires privilege. An experimental `--sif-fuse` flag can be used to perform a mount with `squashfuse` instead, if it is available on your system.
- `--uts` creates a new UTS namespace, which allows a different hostname and/or NIS domain for the container.

To limit presentation of devices from the host into the container, use the `--contain` flag. As well as preventing automatic binds of host directories into the container, `--contain` sets up a minimal `/dev` directory, rather than binding in the entire host `/dev` tree.

Note: When using the `--nv` or `--rocm` flags, GPU devices are present in the container even when `--contain` is used.

Init Shim Process

When a SingularityCE container is run using the `--pid / p` option, or started as an instance (which implies `--pid`), a shim init process is executed that will run the container payload itself.

The shim process helps to ensure signals are propagated correctly from the terminal, or batch schedulers etc. when containers are not designed for interactive use. Because Docker does not provide an init process by default, some containers have been designed to run their own init process, which cannot operate under the control of SingularityCE's shim.

For example, a container using the `tini` init process will produce warnings when started as an instance, or if run with `--pid`. To work around this, use the `--no-init` flag to disable the shim:

```
$ singularity run --pid tini_example.sif
[WARN  tini (2690)] Tini is not running as PID 1 .
Zombie processes will not be re-parented to Tini, so zombie reaping won't work.
To fix the problem, run Tini as PID 1.

$ singularity run --pid --no-init tini_example.sif
...
# NO WARNINGS
```

6.1.6 Docker-like `--compat` Flag

If Docker-like behavior is important, SingularityCE can be started with the `--compat` flag. This flag is a convenient short-hand alternative to using all of:

- `--containall`
- `--no-init`
- `--no-umask`
- `--writable-tmpfs`
- `--no-eval`

A container run with `--compat` has:

- A writable root filesystem, using a temporary overlay where changes are discarded at container exit.
- No automatic bind mounts of `$HOME` or other directories from the host into the container.
- Empty temporary `$HOME` and `/tmp` directories, the contents of which will be discarded at container exit.
- A minimal `/dev` tree, that does not expose host devices inside the container (except GPUs when used with `--nv` or `--rocm`).
- A clean environment, not including environment variables set on the host.
- Its own PID and IPC namespaces.
- No shim init process.
- Argument and environment variable handling matching Docker / OCI runtimes, with respect to evaluation and escaping.

These options will allow most, but not all, Docker / OCI containers to execute correctly under SingularityCE. The user namespace and network namespace are not used, as these negate benefits of SIF and direct access to high performance cluster networks.

Note that behavior in OCI-mode (`--oci`) follows that of `--compat`, by default. To emulate traditional SingularityCE behavior in OCI-mode use the `--no-compat` option.

6.1.7 CMD / ENTRYPOINT Behaviour

When a container is run using `docker`, its default behavior depends on the `CMD` and/or `ENTRYPOINT` set in the `Dockerfile` that was used to build it, along with any arguments on the command line. The `CMD` and `ENTRYPOINT` can also be overridden by flags.

A SingularityCE container has the concept of a *runscript*, which is a single shell script defining what happens when you `singularity run` the container. Because there is no internal concept of `CMD` and `ENTRYPOINT`, SingularityCE must create a runscript from the `CMD` and `ENTRYPOINT` when converting a Docker container. The behavior of this script mirrors Docker as closely as possible.

If the Docker container only has an `ENTRYPOINT` - that `ENTRYPOINT` is run, with any arguments appended:

```
# ENTRYPOINT="date"

# Runs 'date'
$ singularity run mycontainer.sif
Wed 06 Oct 2021 02:42:54 PM CDT
```

(continues on next page)

(continued from previous page)

```
# Runs 'date --utc`
$ singularity run mycontainer.sif --utc
Wed 06 Oct 2021 07:44:27 PM UTC
```

If the Docker container only has a `CMD` - the `CMD` is run, or is *replaced* with any arguments:

```
# CMD="date"

# Runs 'date'
$ singularity run mycontainer.sif
Wed 06 Oct 2021 02:45:39 PM CDT

# Runs 'echo hello'
$ singularity run mycontainer.sif echo hello
hello
```

If the Docker container has a `CMD` *and* `ENTRYPOINT`, then we run `ENTRYPOINT` with either `CMD` as default arguments, or replaced with any user supplied arguments:

```
# ENTRYPOINT="date"
# CMD="--utc"

# Runs 'date --utc'
$ singularity run mycontainer.sif
Wed 06 Oct 2021 07:48:43 PM UTC

# Runs 'date -R'
$ singularity run mycontainer.sif -R
Wed, 06 Oct 2021 14:49:07 -0500
```

There is no flag to override an `ENTRYPOINT` set for a Docker container. Instead, use `singularity exec` to run an arbitrary program inside a container.

Argument Handling

Because SingularityCE runscripts are evaluated shell scripts, arguments can behave slightly differently than in Docker/OCI runtimes if they contain shell code that may be evaluated.

If you are using a container that was directly built or run from a Docker/OCI source, with SingularityCE 3.10 or later, the `--no-eval` flag will prevent this extra evaluation so that arguments are handled in a compatible manner:

```
# docker/OCI behavior
$ docker run -it --rm alpine echo "\$HOSTNAME"
$HOSTNAME

# Singularity default
$ singularity run docker://alpine echo "\$HOSTNAME"
p700

# Singularity with --no-eval
$ singularity run --no-eval docker://alpine echo "\$HOSTNAME"
$HOSTNAME
```

Note: `--no-eval` will not change argument behavior for containers built with SingularityCE 3.9 or earlier, as the handling is implemented in the runscript that is built into the container.

You can check the version of SingularityCE used to build a container with `singularity inspect mycontainer.sif`.

To avoid evaluation without `--no-eval`, and when using containers built with SingularityCE 3.9 or earlier, you will need to add an extra level of shell escaping to arguments on the command line:

```
$ docker run -it --rm alpine echo "\$HOSTNAME"
$HOSTNAME

$ singularity run docker://alpine echo "\$HOSTNAME"
p700

$ singularity run docker://alpine echo "\\\$HOSTNAME"
$HOSTNAME
```

If you are running a binary inside a `docker://` container directly, using the `exec` command, the argument handling mirrors Docker/OCI runtimes as there is no evaluated runscript.

6.1.8 Best Practices for Docker & SingularityCE Compatibility

As detailed previously, SingularityCE can make use of most Docker and OCI images without issues, or via simple workarounds. In general, however, there are some best practices that should be applied when creating Docker / OCI containers that will also be run using SingularityCE.

1. **Don't require execution by a specific user**

Avoid using the `USER` instruction in your Docker file, as it is ignored by Singularity. Install and configure software inside the container so that it can be run by any user.

2. **Don't install software under /root or in another user's home directory**

Because a Docker container builds and runs as the `root` user by default, it's tempting to install software into root's home directory (`/root`). Permissions on `/root` are usually set so that it is inaccessible to non-root users. When the container is run as another user the software may be inaccessible.

Software inside another user's home directory, e.g. `/home/myapp`, may be obscured by SingularityCE's automatic mounts onto `/home`.

Install software into system-wide locations in the container, such as under `/usr` or `/opt` to avoid these issues.

3. **Support a read-only filesystem**

Because of the immutable nature of the SIF format, a container run with SingularityCE is read-only by default.

Try to ensure your container will run with a read-only filesystem. If this is not possible, document exactly where the container needs to write, so that a user can bind in a writable location, or use `--writable-tmpfs` as appropriate.

You can test read-only execution with Docker using `docker run --read-only --tmpfs /run --tmpfs /tmp sylabsio/lolcow`.

4. **Be careful writing to /tmp**

SingularityCE mounts the *host* `/tmp` into the container, by default. This means you must be be careful when writing sensitive information to `/tmp`, and should ensure your container cleans up files it writes there.

5. **Consider library caches / ldconfig**

If your `Dockerfile` adds libraries and / or manipulates the ld search path in the container (`ld.so.conf` / `ld.so.conf.d`), you should ensure the library cache is updated during the build.

Because Singularity runs containers read-only by default, the cache and any missing library symlinks may not be able to be updated / created at execution time.

Run `ldconfig` toward the *end* of your `Dockerfile` to ensure symbolic links and the the `ld.so.cache` are up-to-date.

6.1.9 Troubleshooting

Registry Authentication Issues

If you experience problems pulling containers from a private registry, check your credentials carefully. You can `singularity pull` with the `--docker-login` flag to perform an interactive login. This may be useful if you are unsure whether you have stored credentials properly via `singularity registry login` or `docker login`.

OCI registries expect different values for username and password fields. Some require a token to be generated and used instead of your account password. Some take a generic username, and rely only on the token to identify you. Consult the documentation for your registry carefully. Look for instructions that detail how to login via `docker login` without external helper programs, if possible.

Container Doesn't Start

If a Docker container fails to start, the most common cause is that it needs to write files, while SingularityCE runs read-only by default.

Try running with the `--writable-tmpfs` option, or the `--compat` flag (which enables additional compatibility fixes).

You can also look for error messages mentioning 'permission denied' or 'read-only filesystem'. Note where the program is attempting to write, and use `--bind` or `--mount` to bind a directory from the host system into that location. This will allow the container to write the needed files, which will appear in the directory you bind in.

Unexpected Container Behaviour

If a Docker container runs, but exhibits unexpected behavior, the most likely cause is the different level of isolation that Singularity provides vs Docker.

Try running the container with the `--contain` option, or the `--compat` option (which is more strict). This disables the automatic mount of your home directory, which is a common source of issues where software in the container loads configuration or packages that may be present there.

Getting Help

The community Slack channels and mailing list are excellent places to ask for help with running a specific Docker container. Other users may have already had success running the same container or software. Please don't report issues with specific Docker containers on GitHub, unless you believe they are due to a bug in SingularityCE.

6.1.10 SingularityCE Definition file vs. Dockerfile

An alternative to running Docker containers with SingularityCE is to re-write the `Dockerfile` as a definition file, and build a native SIF image.

The table below gives a quick reference comparing Dockerfile and SingularityCE definition files. For more detail please see *Definition Files*.

SingularityCE Definition file		Dockerfile	
Section	Description	Section	Description
`Bootstrap`	Defines the source of the base image to build your container from. Many bootstrap agents are supported, e.g. `library`, `docker`, `http`, `shub`, `yum`, `debootstrap`.	-	Can only bootstrap from Docker Hub.
`From:`	Specifies the base image from which to the build the container.	`FROM`	Creates a layer from the specified docker image.
`%setup`	Run setup commands outside of the container (on the host system) after the base image bootstrap.	-	Not supported.
`%files`	Copy files from your host to the container, or between build stages.	`COPY`	Copy files from your host to the container, or between build stages.
`%environment`	Declare and set container environment variables.	`ENV`	Declare and set a container environment variable.
`%help`	Provide a help section for your container image.	-	Not supported.
`%post`	Commands that will be run at build-time.	`RUN`	Commands that will be run at build-time.
`%runscript`	Commands that will be run when you `singularity run` the container image.	`ENTRYPOINT CMD`	Commands / arguments that will run on the container image.

6.2 OCI Runtime Support

6.2.1 Overview

The Open Containers Initiative is an independent organization whose mandate is to develop open standards relating to containerization. There are three OCI specifications covering the OCI container image format, distribution methods for containers, and the behaviour of compliant container runtimes.

The OCI specifications inherited from the historic behaviour of Docker, and have been refined over time. The majority of container runtimes and tools, which work with containers on Linux, follow the OCI standards.

SingularityCE was initially developed to address difficulties with using Docker in shared HPC compute environments. Because of the ways these issues were addressed, SingularityCE's default native runtime is not fully OCI compatible. However, over time, SingularityCE has continuously improved compatibility with OCI standards so that the majority of OCI container images can be run using it.

In SingularityCE 4.0, a new OCI-mode is now fully supported. This mode, enabled via the `--oci` CLI option or `oci mode` directive in `singularity.conf`, uses an OCI low-level runtime to execute containers that achieve compatibility that is not possible with SingularityCE's native runtime. The familiar `singularity` command line interface is maintained and unique features such as the SIF image format continue to offer benefits for HPC environments.

OCI compatibility is discussed in three areas of this guide:

1. The *OCI-mode section* of this page introduces the new OCI-mode (`--oci`), which runs OCI / Docker containers using a true OCI low-level runtime.
2. The *Support for Docker* page discusses limitations, compatibility options, and best practices for running OCI / Docker containers with SingularityCE's default runtime.
3. The *OCI Command Group section* of this page documents the `singularity oci` commands, which provide a low-level means to run SingularityCE SIF containers with a command line that matches other OCI runtimes.

6.2.2 OCI Mode (`--oci`)

OCI-mode, enabled with the `--oci` command line option, or the `oci mode` directive in `singularity.conf`, is now fully supported from SingularityCE 4.0. When OCI-mode is enabled:

- OCI containers are executed using `crun` or `runc` as the low-level runtime, for true OCI runtime compatibility.
- Default behaviour (of bind mounts etc.) is comparable to using the `--compat` mode with the native runtime.
- Containers retrieved from OCI sources are encapsulated within a single OCI-SIF file, maintaining the benefits of SIF while avoiding a full conversion into SingularityCE's native container format.

Users are encouraged to employ OCI-mode when their primary use-case for SingularityCE is to run existing containers from Docker Hub or other OCI registries. Behavior will more closely match that described for Docker than with SingularityCE's native runtime.

System Requirements

To use OCI mode, the following requirements must be met by the host system:

- Unprivileged user namespace creation is supported by the kernel, and enabled.
- Subuid and subgid mappings are configured for users who plan to run `--oci` mode.
- The `TMPDIR` / `SINGULARITY_TMPDIR` is located on a filesystem that supports subuid/subgid mapping.
- `crun` or `runc` are available on the `PATH`.

The majority of these requirements are shared with the those of an unprivileged installation of SingularityCE, as OCI mode does not use setuid. See the admin guide for further information on configuring a system appropriately.

Pulling and Running OCI Containers

To activate OCI-mode when running a container from an OCI source (e.g. Docker Hub), add the `--oci` flag to a `run / shell / exec` or `pull` command:

```
# Pull container to an OCI-SIF and run it
$ singularity pull --oci docker://ubuntu
Getting image source signatures
Copying blob 445a6a12be2b done
Copying config c6b84b685f done
Writing manifest to image destination
INFO:    Converting OCI image to OCI-SIF format
INFO:    Squashing image to single layer
INFO:    Writing OCI-SIF image
INFO:    Cleaning up.
$ singularity run --oci ubuntu_latest.oci.sif
dtrudg-sylabs@mini:~$ echo "HELLO OCI WORLD"
HELLO OCI WORLD

# Run directly from a URI
$ singularity exec --oci docker://ubuntu date
INFO:    Using cached OCI-SIF image
Mon Sep  4 12:24:26 UTC 2023
```

Running containers in this manner greatly improves compatibility between SingularityCE's features and the OCI specification. For example, when running in `--oci` mode, Singularity honors the Dockerfile `USER` directive:

```
# I am joeuser outside of the container
$ whoami
joeuser

# The Dockerfile adds a `testuser`
$ cat Dockerfile
FROM alpine
MAINTAINER Joe User
RUN addgroup -g 2000 testgroup
RUN adduser -D -u 2000 -G testgroup testuser
USER testuser
CMD id
```

(continues on next page)

(continued from previous page)

```
# Create and save a docker archive from this Dockerfile
$ docker build --tag docker-user-demo .
$ docker save docker-user-demo > docker-user-demo.tar

# Run the docker archive from singularity
$ singularity run --oci docker-archive:./docker-user-demo.tar
Getting image source signatures
Copying blob 3f8df8c11beb done
Copying blob 78a822fe2a2d done
Copying blob f7cb6364f42b done
Copying config 59af11197a done
Writing manifest to image destination
INFO:    Converting OCI image to OCI-SIF format
INFO:    Squashing image to single layer
INFO:    Writing OCI-SIF image
INFO:    Cleaning up.
uid=2000(testuser) gid=2000(testgroup)
```

As the last line of output shows, the user inside the container run by `singularity run --oci` is `testuser` (the user added as part of the Dockerfile) rather than `joeuser` (the user on the host).

Default Behaviour & `--no-compat`

By and large, the user experience when running in OCI-mode is similar to using the `--compat` flag with the native runtime, or running containers with other tools such as Docker. Particularly:

- A writable in-memory overlay is provided by default. The container can be written to, but changes to the filesystem are lost when the container exits.
- The home directory and current working directory are not mounted into the container.

While these defaults make it simpler to translate `docker run` commands to `singularity run`, OCI-mode can also be used with the `--no-compat` option to emulate SingularityCE's traditional native runtime behaviour:

```
$ singularity shell --oci --no-compat docker://ubuntu

# The container is read-only
Singularity> touch /foo
touch: cannot touch '/foo': Read-only file system

# The current working directory was bind mounted, and is the default entry point
Singularity> pwd
/data

# The user's home directory is bind mounted
Singularity> echo $HOME
/home/example
Singularity> ls $HOME
file1   file2   file3
```

Feature set

With the release of SingularityCE 4.0, the functionality available in OCI mode - that is, when running SingularityCE `shell` / `exec` / `run` commands with the `--oci` flag - is approaching feature-parity with the native SingularityCE runtime, with some important exceptions noted below.

Note: SingularityCE's OCI mode also supports the Container Device Interface (CDI) standard for making GPUs and other devices from the host available inside the container. See the *CDI section*, below, for details.

The following features are supported in `--oci` mode:

- `docker://`, `docker-archive:`, `docker-daemon:`, `oci:`, `oci-archive:`, `library://`, `oras://`, `http://`, and `https://` image sources.
- `--fakeroot` for effective root in the container.
- Bind mounts via `--bind` or `--mount`.
- `--overlay` to mount a SquashFS image (read-only), an EXT3 (read-only or writable), or a directory (read-only or writable), as overlays within the container.
  - Allows changes to the filesystem to persist across runs of the OCI container
  - Multiple simultaneous overlays are supported (though all but one must be mounted as read-only).
- `--cwd` (synonym: `--pwd`) to set a custom starting working-directory for the container.
- `--home` to set the in-container user's home directory. Supplying a single location (e.g. `--home /myhomedir`) will result in a new tmpfs directory being created at the specified location inside the container, and that dir being set as the in-container user's home dir. Supplying two locations separated by a colon (e.g. `--home /home/user:/myhomedir`) will result in the first location on the host being bind-mounted as the second location in-container, and set as the in-container user's home dir.
- `--scratch` (shorthand: `-S`) to mount a tmpfs scratch directory in the container.
- *–workdir* <*workdir*>: if specified, will map */tmp* and */var/tmp* in the container to <*workdir*>*/tmp* and <*workdir*>*/var_tmp*, respectively, on the host (rather than to tmpfs storage, which is the default). If *–scratch* <*scratchdir*> is used in conjunction with *–workdir*, scratch directories will be mapped to subdirectories nested under <*workdir*>*/scratch* on the host, rather than to tmpfs storage.
- `--no-home` to prevent the container home directory from being mounted.
- `--no-mount` to disable the mounting of `proc`, `sys`, `devpts`, `tmp`, and `home` mounts in the container. Note: `dev` cannot be disabled in OCI-mode, and `bind-path` mounts are not supported.
- Support for the `SINGULARITY_CONTAINLIBS` environment variable, to specify libraries to bind into `/.singularity.d/libs/` in the container.
- `--hostname` to set a custom hostname inside the container. (This requires a UTS namespace, therefore this flag will infer `--uts`.)
- Handling `--dns` and `resolv.conf` on a par with native mode: the `--dns` flag can be used to pass a comma-separated list of DNS servers that will be used in the container; if this flag is not used, the container will use the same `resolv.conf` settings as the host.
- Additional namespace requests with `--net`, `--uts`, `--user`.
- `--no-privs` to drop all capabilities from the container process and enable the `NoNewPrivileges` flag.
- `--keep-privs` to keep effective capabilities for the container process (bounding set only for non-root container users).

- `--add-caps` and `--drop-caps`, to modify capabilities of the container process.
- `--rocm` to bind ROCm GPU libraries and devices into the container.
- `--nv` to bind NVIDIA driver / basic CUDA libraries and devices into the container.
- `--apply-cgroups`, and the `--cpu*`, `--blkio*`, `--memory*`, `--pids-limit` flags to apply resource limits.

Features that are not yet supported include but are not limited to:

- Custom `--security` options.
- Support for instances (starting containers in the background).

Running existing non-OCI Singularity containers

OCI-mode can also be used to run containers in SingularityCE's native format, which were created with `singularity build` or pulled without `--oci`. Note that the `--no-compat` option must still be specified to achieve behavior matching the native runtime defaults, otherwise the container will behave as if `--compat` was specified.

When running a native SIF container in OCI-mode, a compatibility warning is shown, as it is not possible to perfectly emulate the behaviour of the native runtime.

```
$ singularity run --oci --no-compat ubuntu_latest.sif
INFO:    Running a non-OCI SIF in OCI mode. See user guide for compatibility information.
```

Environment and action scripts are run using the container's shell, rather than an embedded shell interpreter. Complex environment scripts may exhibit different behavior. Bare images that do not contain `/bin/sh` cannot be run.

OCI-SIF Images

When an OCI container is `pull`-ed or run directly from a URI in OCI-mode, it is encapsulated within a single OCI-SIF file.

Note: OCI-SIF files are only supported by SingularityCE 4.0 and above. They cannot be run using older versions of SingularityCE.

An OCI-SIF file provides the same benefits as a native SIF images:

- The single file is easy to share and transport between systems, including air-gapped hosts.
- The container root filesystem is mounted at the point of execution, directly from the SIF. This avoids excessive metadata traffic when images are stored on shared network filesystems.
- Digital signatures may be added to the OCI-SIF, and later verifed, using the `singularity sign` and `singularity verify` commands.

An OCI-SIF differs from a native-runtime SIF as it aims to minimize the ways in which the encapsulated container differs from its source:

- Container configuration and files are stored as OCI Blobs. This means that the container can be pushed from the OCI-SIF to a registry as a native OCI image, consisting of these blobs, rather than an ORAS entry / artifact.
- The container's OCI image manifest and config are preserved, and directly inserted into the OCI-SIF.
- The container's root filesystem is squashed to a single layer, in squashfs format, but using an approach that better preserves metadata than for native SIF images.
- The container root filesystem is not modified by the addition of Singularity-specific files and directories.

The differences in the layout of native SIF and OCI-SIF images can be seen by inspecting the same OCI container, pulled with and without the `--oci` flag.

```
$ singularity pull docker://ubuntu

$ singularity sif list ubuntu_latest.sif
--------------------------------------------------------------------------------
ID   |GROUP   |LINK    |SIF POSITION (start-end)  |TYPE
--------------------------------------------------------------------------------
1    |1       |NONE    |32176-32208               |Def.FILE
2    |1       |NONE    |32208-35190               |JSON.Generic
3    |1       |NONE    |35190-35386               |JSON.Generic
4    |1       |NONE    |36864-29818880            |FS (Squashfs/*System/amd64)
```

The native SIF, shown above, includes SingularityCE specific entries, such as a definition file and metadata.

```
$ singularity pull --oci docker://ubuntu

$ singularity sif list ubuntu_latest.oci.sif
--------------------------------------------------------------------------------
ID   |GROUP   |LINK    |SIF POSITION (start-end)  |TYPE
--------------------------------------------------------------------------------
1    |1       |NONE    |32176-29806000            |OCI.Blob
2    |1       |NONE    |29806000-29806807         |OCI.Blob
3    |1       |NONE    |29806807-29807216         |OCI.Blob
4    |1       |NONE    |29807216-29807456         |OCI.RootIndex
```

The OCI-SIF contains three `OCI.Blob` entries. These are the container root filesystem (as a single squashfs format layer), the image config, and the image manifest, respectively. There is no definition file or Singularity specific JSON metadata.

The final `OCI.RootIndex` is for internal use, and indexes the content of the OCI-SIF.

Future Development

Subsequent 4.x releases of SingularityCE will aim to improve OCI-mode to address missing features vs native mode, and improve the utility of the OCI-SIF format. Development items include:

- Support for overlays embedded in OCI-SIF files.
- Multi-layer OCI-SIF images, allowing full preservation of the structure of an OCI image embedded in an OCI-SIF. This will permit faithful 2-way translation to/from OCI-SIF.

6.2.3 Container Device Interface (CDI)

Beginning in SingularityCE 4.0, `--oci` mode supports the Container Device Interface (CDI) standard for making GPUs and other devices from the host available inside the container. It offers an alternative to previous approaches that were vendor specific, and unevenly supported across different container runtimes. Users of NVIDIA GPUs, and other devices with CDI configurations, will benefit from a consistent way of using them in containers that spans the cloud native and HPC fields.

SingularityCE's "action" commands (`run` / `exec` / `shell`), when run in OCI mode, now support a `--device` flag:

```
--device strings                fully-qualified CDI device name(s).
                                A fully-qualified CDI device name
                                consists of a VENDOR, CLASS, and
                                NAME, which are combined as follows:
                                <VENDOR>/<CLASS>=<NAME> (e.g.
                                vendor.com/device=mydevice).
                                Multiple fully-qualified CDI device
                                names can be given as a comma
                                separated list.
```

This allows device from the host to be mapped into the container with the added benefits of the CDI standard, including:

- Exposing multiple nodes on `/dev` as part of what is, notionally, a single "device".
- Mounting files from the runtime namespace required to support the device.
- Hiding procfs entries.
- Performing compatibility checks between the container and the device to determine whether to make it available in-container.
- Performing runtime-specific operations (e.g. VM vs Linux container-based runtimes).
- Performing device-specific operations (e.g. scrubbing the memory of a GPU or reconfiguring an FPGA).

In addition, SingularityCE's OCI mode provides a `--cdi-dirs` flag, which enables the user to override the default search directory for CDI definition files:

```
--cdi-dirs strings              comma-separated list of directories
                                in which CDI should look for device
                                definition JSON files. If omitted,
                                default will be: /etc/cdi,/var/run/cdi
```

6.2.4 OCI Command Group

To support execution of containers via a CLI conforming to the OCI runtime specification, Singularity provides the `oci` command group.

The `oci` command group is not intended for end users, but as a low-level interface that can be leveraged by other software. In most circumstances, the OCI-mode (`--oci`) should be used instead of the `oci` command group.

Note: All commands in the `oci` command group currently require `root` privileges.

OCI containers follow a different lifecycle from containers that are run with `singularity run/shell/exec`. Rather than being a simple process that starts, and exits, they are created, run, killed, and deleted. This is similar to instances. Additionally, containers must be run from an OCI bundle, which is a specific directory structure that holds the container's root filesystem and configuration file. To run a SingularityCE SIF image, you must mount it into a bundle.

Mounting an OCI Filesystem Bundle

Let's work with a busybox container image, pulling it down with the default `busybox_latest.sif` filename:

```
$ singularity pull library://busybox
INFO:    Downloading library image
773.7KiB / 773.7KiB [============================================================]␣
↪100 % 931.4 KiB/s 0s
```

Now use `singularity oci mount` to create an OCI bundle onto which the SIF is mounted:

```
$ sudo singularity oci mount ./busybox_latest.sif /var/tmp/busybox
```

By issuing the `mount` command, the root filesystem encapsulated in the SIF file `busybox_latest.sif` is mounted on `/var/tmp/busybox` with an overlay setup to hold any changes, as the SIF file is read-only.

Content of an OCI Compliant Filesystem Bundle

The OCI bundle, created by the mount command consists of the following files and directories:

- `config.json` - a generated OCI container configuration file, which instructs the OCI runtime how to run the container, which filesystems to bind mount, what environment to set, etc.
- `overlay/` - a directory that holds the contents of the bundle overlay - any new files, or changed files, that differ from the content of the read-only SIF container image.
- `rootfs/` - a directory containing the mounted root filesystem from the SIF container image, with its overlay.
- `volumes/` - a directory used by the runtime to stage any data mounted into the container as a volume.

OCI config.json

The container configuration file, `config.json` in the OCI bundle, is generated by `singularity mount` with generic default options. It may not reflect the `config.json` used by an OCI runtime working directly from a native OCI image, rather than a mounted SIF image.

You can inspect and modify `config.json` according to the OCI runtime specification to influence the behavior of the container.

Running a Container

For simple interactive use, the `oci run` command will create and start a container instance, attaching to it in the foreground. This is similar to the way `singularity run` works, with SingularityCE's native runtime engine:

```
$ sudo singularity oci run -b /var/tmp/busybox busybox1
/ # echo "Hello"
Hello
/ # exit
```

When the process running in the container (in this case a shell) exits, the container is automatically cleaned up, but note that the OCI bundle remains mounted.

Full Container Lifecycle

If you want to run a detached background service, or interact with SIF containers from 3rd party tools that are compatibile with OCI runtimes, you will step through the container lifecycle using a number of `oci` subcommands. These move the container between different states in the lifecycle.

Once an OCI bundle is available, you can create a instance of the container with the `oci create` subcommand:

```
$ sudo singularity oci create -b /var/tmp/busybox busybox1
INFO:    Container busybox1 created with PID 20105
```

At this point the runtime has prepared container processes, but the payload (`CMD / ENTRYPOINT` or `runscript`) has not been started.

Check the state of the container using the `oci state` subcommand:

```
$ sudo singularity oci state busybox1
{
  "ociVersion": "1.0.2-dev",
  "id": "busybox1",
  "pid": 20105,
  "status": "created",
  "bundle": "/var/tmp/busybox",
  "rootfs": "/var/tmp/busybox/rootfs",
  "created": "2022-04-27T15:39:08.751705502Z",
  "owner": ""
}
```

Start the container's `CMD/ENTRYPOINT` or `runscript` with the `oci start` command:

```
$ singularity start busybox1
```

There is no output, but if you check the container state it will now be `running`. The container is *detached*. To view output or provide input we will need to attach to its input and output streams. with the `oci attach` command:

```
$ sudo singularity oci attach busybox1
/ # date
date
Wed Apr 27 15:45:27 UTC 2022
/ #
```

When finished with the container, first `oci kill` running processes, than `oci delete` the container instance:

```
$ sudo singularity oci kill busybox1
$ sudo singularity oci delete busybox1
```

Unmounting OCI Filesystem Bundles

When you are finished with an OCI bundle, you will need to explicitly unmount it using the `oci umount` subcommand:

```
$ sudo singularity oci umount /var/tmp/busybox
```

Technical Implementation

SingularityCE 3.10 and newer use runc as the low-level runtime engine to execute containers in an OCI Runtime Spec compliant manner. `runc` is expected to be provided by your Linux distribution.

To manage container i/o streams and attachment, conmon is used. SingularityCE ships with a suitable version of *conmon* to support the `oci` command group.

In SingularityCE 3.9 and prior, SingularityCE's own low-level runtime was employed for `oci` operations. This was retired to simplify maintenance, improve OCI compliance, and facilitate the development of OCI-mode.

6.2.5 OCI Specification Coverage

OCI Image Spec

In the default native mode, SingularityCE can convert container images that satisfy the OCI Image Format Specification into its own native SIF format or a simple sandbox directory. Most of the configuration that a container image can specify is supported by the SingularityCE runtime, but there are *various limitations* and workarounds may be required for some containers.

In OCI-mode, SingularityCE encapsulates OCI images in an OCI-SIF. The image config is preserved, and the container runs via a low-level OCI runtime for compatibiility with features of the image specification.

OCI Distribution Spec

SingularityCE is able to pull images from registries that satisfy the OCI Distribution Specification.

Native SIF images can be pushed, as a single file, to registries that permit artifacts with arbitrary content types using `oras://` URIs.

OCI-SIF images can be pushed to registries as a single file with `oras://` URIs, or as an OCI image with `docker://` URIs.

Note: Although OCI-SIF images can be pushed to a registry as an OCI image, the squashfs layer format is not currently supported by other runtimes. The images can only be retrieved and run by SingularityCE 4.

OCI Runtime Spec

In its default mode, using the native runtime, SingularityCE does not follow the OCI Runtime Specification closely. Instead, it uses its own runtime that is better matched to the requirements and limitations of multi-user shared compute environments. The `--compat` option will apply various other flags to achieve behaviour that is closer (but not identical) to OCI runtimes such as Docker.

In OCI-mode, SingularityCE executes containers with a true OCI compatible low-level runtime. This allows compatibility with features of the OCI runtime specification that are not possible with the native runtime.

OCI Runtime CLI

The `singularity oci` commands were added to provide a mode of operation in which SingularityCE does implement the OCI runtime specification and container lifecycle, via a command line compatible with other low-level OCI runtimes. These commands are primarily of interest to tooling that might use SingularityCE as a container runtime, rather than end users. End users will general use the OCI-mode (`--oci`) with `run / shell / exec`.

6.3 SingularityCE and MPI applications

The Message Passing Interface (MPI) is a standard extensively used by HPC applications to implement various communication across compute nodes of a single system or across compute platforms. There are two main open-source implementations of MPI at the moment - OpenMPI and MPICH, both of which are supported by SingularityCE. The goal of this page is to demonstrate the development and running of MPI programs using SingularityCE containers.

There are several ways of carrying this out, the most popular way of executing MPI applications installed in a SingularityCE container is to rely on the MPI implementation available on the host. This is called the *Host MPI* or the *Hybrid* model since both the MPI implementations provided by system administrators (on the host) and in the containers will be used.

Another approach is to only use the MPI implementation available on the host and not include any MPI in the container. This is called the *Bind* model since it requires to bind/mount the MPI version available on the host into the container.

Note: The *bind* model requires users to be able to mount user-specified files from the host into the container. This ability is sometimes disabled by system administrators for operational reasons. If this is the case on your system please follow the *hybrid* approach.

6.3.1 Hybrid model

The basic idea behind the *Hybrid Approach* is when you execute a SingularityCE container with MPI code, you will call `mpiexec` or a similar launcher on the `singularity` command itself. The MPI process outside of the container will then work in tandem with MPI inside the container and the containerized MPI code to instantiate the job.

The Open MPI/SingularityCE workflow in detail:

1. The MPI launcher (e.g., `mpirun`, `mpiexec`) is called by the resource manager or the user directly from a shell.
2. Open MPI then calls the process management daemon (ORTED).
3. The ORTED process launches the SingularityCE container requested by the launcher command.
4. SingularityCE instantiates the container and namespace environment.
5. SingularityCE then launches the MPI application within the container.

6. The MPI application launches and loads the Open MPI libraries.
7. The Open MPI libraries connect back to the ORTED process via the Process Management Interface (PMI).

At this point the processes within the container run as they would normally directly on the host.

The advantages of this approach are:

- Integration with resource managers such as Slurm.
- Simplicity since similar to natively running MPI applications.

The drawbacks are:

- The MPI in the container must be compatible with the version of MPI available on the host.
- The MPI implementation in the container must be carefully configured for optimal use of the hardware if performance is critical.

Since the MPI implementation in the container must be compliant with the version available on the host system, a standard approach is to build your own MPI container, including building the same MPI framework installed on the host from source.

Test Application

To illustrate how SingularityCE can be used to execute MPI applications, we will assume for a moment that the application is `mpitest.c`, a simple Hello World:

```
#include <mpi.h>
#include <stdio.h>
#include <stdlib.h>

int main (int argc, char **argv) {
        int rc;
        int size;
        int myrank;

        rc = MPI_Init (&argc, &argv);
        if (rc != MPI_SUCCESS) {
                fprintf (stderr, "MPI_Init() failed");
                return EXIT_FAILURE;
        }

        rc = MPI_Comm_size (MPI_COMM_WORLD, &size);
        if (rc != MPI_SUCCESS) {
                fprintf (stderr, "MPI_Comm_size() failed");
                goto exit_with_error;
        }

        rc = MPI_Comm_rank (MPI_COMM_WORLD, &myrank);
        if (rc != MPI_SUCCESS) {
                fprintf (stderr, "MPI_Comm_rank() failed");
                goto exit_with_error;
        }

        fprintf (stdout, "Hello, I am rank %d/%d\n", myrank, size);
```

(continues on next page)

(continued from previous page)

```
        MPI_Finalize();

        return EXIT_SUCCESS;

 exit_with_error:
        MPI_Finalize();
        return EXIT_FAILURE;
}
```

Note: MPI is an interface to a library, so it consists of function calls and libraries that can be used by many programming languages. It comes with standardized bindings for Fortran and C. However, it can support applications in many languages like Python, R, etc.

The next step is to create the definition file used to build the container, which will depend on the MPI implementation available on the host.

MPICH Hybrid Container

If the host MPI is MPICH, a definition file such as the following example can be used:

```
Bootstrap: docker
From: ubuntu:22.04

%files
    mpitest.c /opt

%environment
    # Point to MPICH binaries, libraries man pages
    export MPICH_DIR=/opt/mpich
    export PATH="$MPICH_DIR/bin:$PATH"
    export LD_LIBRARY_PATH="$MPICH_DIR/lib:$LD_LIBRARY_PATH"
    export MANPATH=$MPICH_DIR/share/man:$MANPATH

%post
    echo "Installing required packages..."
    export DEBIAN_FRONTEND=noninteractive
    apt-get update && apt-get install -y wget git bash gcc gfortran g++ make python3-dev

    # Information about the version of MPICH to use
    export MPICH_VERSION=4.1.1
    export MPICH_URL="http://www.mpich.org/static/downloads/$MPICH_VERSION/mpich-$MPICH_
↪VERSION.tar.gz"
    export MPICH_DIR=/opt/mpich

    echo "Installing MPICH..."
    mkdir -p /tmp/mpich
    mkdir -p /opt
    # Download
    cd /tmp/mpich && wget -O mpich-$MPICH_VERSION.tar.gz $MPICH_URL && tar xzf mpich-
↪$MPICH_VERSION.tar.gz
```

(continues on next page)

(continued from previous page)

```
    # Compile and install
    cd /tmp/mpich/mpich-$MPICH_VERSION && ./configure --prefix=$MPICH_DIR && make -j
↪$(nproc) install

    # Set env variables so we can compile our application
    export PATH=$MPICH_DIR/bin:$PATH
    export LD_LIBRARY_PATH=$MPICH_DIR/lib:$LD_LIBRARY_PATH

    echo "Compiling the MPI application..."
    cd /opt && mpicc -o mpitest mpitest.c
```

Note: The version of MPICH you install in the container must be compatible with the version on the host. It should also be configured to support the same process management mechanism and version, e.g. PMI2 / PMIx, as used on the host.

There are wide variations in MPI configuration across HPC systems. Consult your system documentation, or ask your support staff for details.

Open MPI Hybrid Container

If the host MPI is Open MPI, the definition file looks like:

```
Bootstrap: docker
From: ubuntu:22.04

%files
    mpitest.c /opt

%environment
    # Point to OMPI binaries, libraries, man pages
    export OMPI_DIR=/opt/ompi
    export PATH="$OMPI_DIR/bin:$PATH"
    export LD_LIBRARY_PATH="$OMPI_DIR/lib:$LD_LIBRARY_PATH"
    export MANPATH="$OMPI_DIR/share/man:$MANPATH"

%post
    echo "Installing required packages..."
    apt-get update && apt-get install -y wget git bash gcc gfortran g++ make file bzip2

    echo "Installing Open MPI"
    export OMPI_DIR=/opt/ompi
    export OMPI_VERSION=4.1.5
    export OMPI_URL="https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-$OMPI_
↪VERSION.tar.bz2"
    mkdir -p /tmp/ompi
    mkdir -p /opt
    # Download
    cd /tmp/ompi && wget -O openmpi-$OMPI_VERSION.tar.bz2 $OMPI_URL && tar -xjf openmpi-
↪$OMPI_VERSION.tar.bz2
    # Compile and install
```

(continues on next page)

(continued from previous page)

```
    cd /tmp/ompi/openmpi-$OMPI_VERSION && ./configure --prefix=$OMPI_DIR && make -j
↪$(nproc) install

    # Set env variables so we can compile our application
    export PATH=$OMPI_DIR/bin:$PATH
    export LD_LIBRARY_PATH=$OMPI_DIR/lib:$LD_LIBRARY_PATH

    echo "Compiling the MPI application..."
    cd /opt && mpicc -o mpitest mpitest.c
```

Note: The version of Open MPI you install in the container must be compatible with the version on the host. It should also be configured to support the same process management mechanism and version, e.g. PMI2 / PMIx, as used on the host.

There are wide variations in MPI configuration across HPC systems. Consult your system documentation, or ask your support staff for details.

Running an MPI Application

The standard way to execute MPI applications with hybrid SingularityCE containers is to run the native `mpirun` command from the host, which will start SingularityCE containers and ultimately MPI ranks within the containers.

Assuming your container with MPI and your application is already built, the `mpirun` command to start your application looks like when your container has been built based on the hybrid model:

```
$ mpirun -n <NUMBER_OF_RANKS> singularity exec <PATH/TO/MY/IMAGE> </PATH/TO/BINARY/
↪WITHIN/CONTAINER>
```

Practically, this command will first start a process instantiating `mpirun` and then SingularityCE containers on compute nodes. Finally, when the containers start, the MPI binary is executed:

```
$ mpirun -n 8 singularity run hybrid-mpich.sif /opt/mpitest
Hello, I am rank 3/8
Hello, I am rank 4/8
Hello, I am rank 6/8
Hello, I am rank 2/8
Hello, I am rank 0/8
Hello, I am rank 5/8
Hello, I am rank 1/8
Hello, I am rank 7/8
```

6.3.2 Bind model

Similar to the *Hybrid Approach*, the basic idea behind the *Bind Approach* is to start the MPI application by calling the MPI launcher (e.g., *mpirun*) from the host. The main difference between the hybrid and bind approach is the fact that with the bind approach, the container usually does not include any MPI implementation. This means that SingularityCE needs to mount/bind the MPI available on the host into the container.

Technically this requires two steps:

1. Know where the MPI implementation on the host is installed.
2. Mount/bind it into the container in a location where the system will be able to find libraries and binaries.

The advantages of this approach are:

- Integration with resource managers such as Slurm.
- Container images are smaller since there is no need to add an MPI in the containers.

The drawbacks are:

- The MPI used to compile the application in the container must be compatible with the version of MPI available on the host.
- The user must know where the host MPI is installed.
- The user must ensure that binding the directory where the host MPI is installed is possible.
- The user must ensure that the host MPI is compatible with the MPI used to compile and install the application in the container.

The creation of a SingularityCE container for the bind model is based on the following steps:

1. Compile your application on a system with the target MPI implementation, as you would do to install your application on any system.
2. Create a definition file that includes the copy of the application from the host to the container image, as well as all required dependencies.
3. Generate the container image.

As already mentioned, the compilation of the application on the host is not different from the installation of your application on any system. Just make sure that the MPI on the system where you create your container is compatible with the MPI available on the platform(s) where you want to run your containers. For example, a container where the application has been compiled with MPICH will not be able to run on a system where only Open MPI is available, even if you mount the directory where Open MPI is installed.

Bind Mode Definition File

A definition file for a container in bind mode is fairly straight forward. The following example shows the definition file for the test program, which in this case has been compiled on the host to `/tmp/mpitest`:

```
Bootstrap: docker
From: ubuntu:22.04

%files
      /tmp/mpitest /opt/mpitest

%environment
      export PATH="$MPI_DIR/bin:$PATH"
      export LD_LIBRARY_PATH="$MPI_DIR/lib:$LD_LIBRARY_PATH"
```

In this example, the application `mpitest` is copied from the host into `/opt`, so we will need to run it as `/opt/mpitest` inside our container.

The environment section adds paths for binaries and libraries under `$MPI_DIR` - which we will need to set when running the container.

Running an MPI Application

When running our bind mode container we need to `--bind` our host's MPI installation into the container. We also need to set the environment variable `$MPI_DIR` in the container to point to the location where the MPI installation is bound in.

Setting up the container in this way makes it semi-portable between systems that have a version-compatible MPI installation, but under different installation paths. You can also hard code the MPI path in the definition file if you wish.

```
$ export MPI_DIR="<PATH/TO/HOST/MPI/DIRECTORY>"
$ mpirun -n <NUMBER_OF_RANKS> singularity exec --bind "$MPI_DIR" <PATH/TO/MY/IMAGE> </
↪PATH/TO/BINARY/WITHIN/CONTAINER>
```

On an example system we may be using an Open MPI installation at `/cm/shared/apps/openmpi/gcc/64/4.0.5/`. This means that the commands to run the container in bind mode are:

```
$ export MPI_DIR="/cm/shared/apps/openmpi/gcc/64/4.0.5"
$ mpirun -n 8 singularity exec --bind "$MPI_DIR" bind.sif /opt/mpitest
Hello, I am rank 1/8
Hello, I am rank 2/8
Hello, I am rank 0/8
Hello, I am rank 7/8
Hello, I am rank 5/8
Hello, I am rank 3/8
Hello, I am rank 4/8
Hello, I am rank 6/8
```

6.3.3 Batch Scheduler / Slurm

If your target system is setup with a batch system such as SLURM, a standard way to execute MPI applications is through a batch script. The following example illustrates the context of a batch script for Slurm that aims at starting a SingularityCE container on each node allocated to the execution of the job. It can easily be adapted for all major batch systems available.

```
$ cat my_job.sh
#!/bin/bash
#SBATCH --job-name singularity-mpi
#SBATCH -N $NNODES # total number of nodes
#SBATCH --time=00:05:00 # Max execution time

mpirun -n $NP singularity exec /var/nfsshare/gvallee/mpich.sif /opt/mpitest
```

In fact, the example describes a job that requests the number of nodes specified by the `NNODES` environment variable and a total number of MPI processes specified by the `NP` environment variable. The example is also assuming that the container is based on the hybrid model; if it is based on the bind model, please add the appropriate bind options.

A user can then submit a job by executing the following SLURM command:

```
$ sbatch my_job.sh
```

6.3.4 Alternative Launchers

On many systems it is common to use an alternative launcher to start MPI applications, e.g. Slurm's `srun` rather than the `mpirun` provided by the MPI installation. This approach is supported with SingularityCE as long as the container MPI version supports the same process management interface (e.g. PMI2 / PMIx) and version as is used by the launcher.

In the bind mode the host MPI is used in the container, and should interact correctly with the same launchers as it does on the host.

6.3.5 Interconnects / Networking

High performance interconnects such as Infiniband and Omnipath require that MPI implementations are built to support them. You may need to install or bind Infiniband/Omnipath libraries into your containers when using these interconnects.

By default SingularityCE exposes every device in `/dev` to the container. If you run a container using the `--contain` or `--containall` flags a minimal `/dev` is used instead. You may need to bind in additional `/dev/` entries manually to support the operation of your interconnect drivers in the container in this case.

6.3.6 Troubleshooting Tips

If your containers run N rank 0 processes, instead of operating correctly as an MPI application, it is likely that the MPI stack used to launch the containerized application is not compatible with, or cannot communicate with, the MPI stack in the container.

E.g. if we attempt to run the hybrid Open MPI container, but with `mpirun` from MPICH loaded on the host:

```
$ module add mpich
$ mpirun -n 8 singularity run hybrid-openmpi.sif /opt/mpitest
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
Hello, I am rank 0/1
```

If your container starts processes of different ranks, but fails with communications errors there may also be a version incompatibility, or interconnect libraries may not be available or configured properly with the MPI stack in the container.

Please check the following things carefully before asking questions in the SingularityCE community:

- For the hybrid mode, is the MPI version on the host compatible with the version in the container? Newer MPI versions can generally tolerate some mismatch in the version number, but it is safest to use identical versions.
- Is the MPI stack in the container configured to support the process management method used on the host? E.g. if you are launching tasks with `srun` configured for PMIx only, then a containerized MPI supporting PMI2 only will not operate as expected.
- If you are using an interconnect other than standard Ethernet, are any required libraries for it installed or bound into the container? Is the MPI stack in the container configured correctly to use them?

We recommend using the SingularityCE Google Group or Slack Channel to ask for MPI advice from the SingularityCE community. HPC cluster configurations vary greatly and most MPI problems are related to MPI / interconnect configuration, and not caused by issues in SingularityCE itself.

6.4 GPU Support (NVIDIA CUDA & AMD ROCm)

SingularityCE natively supports running application containers that use NVIDIA's CUDA GPU compute framework, or AMD's ROCm solution. This allows easy access to users of GPU-enabled machine learning frameworks such as tensorflow, regardless of the host operating system. As long as the host has a driver and library installation for CUDA/ROCm then it's possible to e.g. run tensorflow in an up-to-date Ubuntu 20.04 container, from an older RHEL 7 host.

Applications that support OpenCL for compute acceleration can also be used easily, with an additional bind option.

With SingularityCE 3.9 experimental support has been introduced to use NVIDIA's `nvidia-container-cli` tooling for GPU container setup. This functionality, accessible via the new `--nvccli` flag, improves compatibility with OCI runtimes and exposes additional container configuration options.

6.4.1 NVIDIA GPUs & CUDA (Legacy)

Commands that `run`, or otherwise execute containers (`shell`, `exec`) can take an `--nv` option, which will setup the container's environment to use an NVIDIA GPU and the basic CUDA libraries to run a CUDA enabled application. The `--nv` flag will:

- Ensure that the `/dev/nvidiaX` device entries are available inside the container, so that the GPU cards in the host are accessible.
- Locate and bind the basic CUDA libraries from the host into the container, so that they are available to the container, and match the kernel GPU driver on the host.
- Set the `LD_LIBRARY_PATH` inside the container so that the bound-in version of the CUDA libraries are used by applications run inside the container.

Requirements

To use the `--nv` flag to run a CUDA application inside a container you must ensure that:

- The host has a working installation of the NVIDIA GPU driver, and a matching version of the basic NVIDIA/CUDA libraries. The host *does not* need to have an X server running, unless you want to run graphical apps from the container.
- The NVIDIA libraries are in the system's library search path.
- The application inside your container was compiled for a CUDA version, and device capability level, that is supported by the host card and driver.

These requirements are usually satisfied by installing the NVIDIA drivers and CUDA packages directly from the NVIDIA website. Linux distributions may provide NVIDIA drivers and CUDA libraries, but they are often outdated which can lead to problems running applications compiled for the latest versions of CUDA.

SingularityCE will find the NVIDIA/CUDA libraries on your host using the list of libraries in the configuration file `etc/singularity/nvbliblist`, and resolving paths through the `ldconfig` cache. At time of release this list is appropriate for the latest stable CUDA version. It can be modified by the administrator to add additional libraries if necessary. See the admin guide for more details.

Example - tensorflow-gpu

Tensorflow is commonly used for machine learning projects but can be difficult to install on older systems, and is updated frequently. Running tensorflow from a container removes installation problems and makes trying out new versions easy.

The official tensorflow repository on Docker Hub contains NVIDA GPU supporting containers, that will use CUDA for processing. You can view the available versions on the tags page on Docker Hub

The container is large, so it's best to build or pull the docker image to a SIF before you start working with it:

```
$ singularity pull docker://tensorflow/tensorflow:latest-gpu
...
INFO:    Creating SIF file...
INFO:    Build complete: tensorflow_latest-gpu.sif
```

Then run the container with GPU support:

```
$ singularity run --nv tensorflow_latest-gpu.sif


________                                _________________
___  __/__________________________________  ____/__  /________      __
__  /  _  _ \_  __ \_  ___/  __ \_  ___/_  /_   __  /_  __ \_ | /| / /
_  /   /  __/  / / /(__  )/ /_/ /  /   _  __/   _  / / /_/ /_ |/ |/ /
/_/    \___//_/ /_//____/ \____//_/    /_/      /_/  \____/____/|__/


You are running this container as user with ID 1000 and group 1000,
which should map to the ID and group for your user on the Docker host. Great!

Singularity>
```

You can verify the GPU is available within the container by using the tensorflow `list_local_devices()` function:

```
Singularity> python
Python 2.7.15+ (default, Jul  9 2019, 16:51:35)
[GCC 7.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from tensorflow.python.client import device_lib
>>> print(device_lib.list_local_devices())
2019-11-14 15:32:09.743600: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your␣
↪CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
2019-11-14 15:32:09.784482: I tensorflow/core/platform/profile_utils/cpu_utils.cc:94]␣
↪CPU Frequency: 3292620000 Hz
2019-11-14 15:32:09.787911: I tensorflow/compiler/xla/service/service.cc:168] XLA␣
↪service 0x565246634360 executing computations on platform Host. Devices:
2019-11-14 15:32:09.787939: I tensorflow/compiler/xla/service/service.cc:175]  ␣
↪StreamExecutor device (0): Host, Default Version
2019-11-14 15:32:09.798428: I tensorflow/stream_executor/platform/default/dso_loader.
↪cc:44] Successfully opened dynamic library libcuda.so.1
2019-11-14 15:32:09.842683: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:1006]␣
↪successful NUMA node read from SysFS had negative value (-1), but there must be at␣
↪least one NUMA node, so returning NUMA node zero
2019-11-14 15:32:09.843252: I tensorflow/compiler/xla/service/service.cc:168] XLA␣
↪service 0x5652469263d0 executing computations on platform CUDA. Devices:
```

(continues on next page)

(continued from previous page)

```
2019-11-14 15:32:09.843265: I tensorflow/compiler/xla/service/service.cc:175]  ␣
↪StreamExecutor device (0): GeForce GT 730, Compute Capability 3.5
2019-11-14 15:32:09.843380: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:1006]␣
↪successful NUMA node read from SysFS had negative value (-1), but there must be at␣
↪least one NUMA node, so returning NUMA node zero
2019-11-14 15:32:09.843984: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1618]␣
↪Found device 0 with properties:
name: GeForce GT 730 major: 3 minor: 5 memoryClockRate(GHz): 0.9015
...
```

Multiple GPUs

By default, SingularityCE makes all host devices available in the container. When the `--contain` option is used a minimal `/dev` tree is created in the container, but the `--nv` option will ensure that all nvidia devices on the host are present in the container.

This behaviour is different to `nvidia-docker` where an `NVIDIA_VISIBLE_DEVICES` environment variable is used to control whether some or all host GPUs are visible inside a container. The `nvidia-container-runtime` explicitly binds the devices into the container dependent on the value of `NVIDIA_VISIBLE_DEVICES`.

To control which GPUs are used in a SingularityCE container that is run with `--nv` you can set `SINGULARITYENV_CUDA_VISIBLE_DEVICES` before running the container, or `CUDA_VISIBLE_DEVICES` inside the container. This variable will limit the GPU devices that CUDA programs see.

E.g. to run the tensorflow container, but using only the first GPU in the host, we could do:

```
$ SINGULARITYENV_CUDA_VISIBLE_DEVICES=0 singularity run --nv tensorflow_latest-gpu.sif

# or

$ export SINGULARITYENV_CUDA_VISIBLE_DEVICES=0
$ singularity run tensorflow_latest-gpu.sif
```

Troubleshooting

If the host installation of the NVIDIA / CUDA driver and libraries is working and up-to-date there are rarely issues running CUDA programs inside of SingularityCE containers. The most common issue seen is:

CUDA_ERROR_UNKNOWN when everything seems to be correctly configured

CUDA depends on multiple kernel modules being loaded. Not all of the modules are loaded at system startup. Some portions of the NVIDA driver stack are initialized when first needed. This is done using a setuid root binary, so initializing can be triggered by any user on the host. In SingularityCE containers, privilege escalation is blocked, so the setuid root binary cannot initialize the driver stack fully.

If you experience `CUDA_ERROR_UNKNOWN` in a container, initialize the driver stack on the host first, by running a CUDA program there or `modprobe nvidia_uvm` as root, and using `nvidia-persistenced` to avoid driver unload.

6.4.2 NVIDIA GPUs & CUDA (nvidia-container-cli)

SingularityCE 3.9 introduces the `--nvccli` option, which will instruct SingularityCE to perform GPU container setup using the `nvidia-container-cli` utility. This utility must be installed separately from SingularityCE. It is available in the repositories of some distributions, and at: https://nvidia.github.io/libnvidia-container/

> **Warning:** This feature is considered experimental in SingularityCE 3.9. It cannot not replace the legacy NVIDIA support in all situations, and should be tested carefully before use in production workflows.

Using `nvidia-container-cli` to configure a container for GPU operation has a number of advantages, including:

- The tool is maintained by NVIDIA, and will track new features / libraries in new CUDA releases closely.
- Support for passing only specific GPUs / MIG devices into the container.
- Support for providing different classes of GPU capability to the container, e.g. compute, graphics, and display functionality.
- Configuration via the same environment variables that are in use with OCI containers.

Requirements & Limitations

- `nvidia-container-cli` must be installed on your host, owned by the `root` user. Its path must be set in `singularity.conf`. This value will be set at build time if `nvidia-container-cli` is found on the search `$PATH`.
- Your system should support the `overlay` filesystem if you will be running SIF containers in set-uid mode.
- `--nvccli` cannot currently be used in combination with `--fakeroot` in a set-uid install of Singularity. Use the traditional binding method with `--nv` only.
- There are known problems with library discovery for the current `nvidia-container-cli` in recent Debian distributions. See this GitHub issue

Example - tensorflow-gpu

Tensorflow can be run using `--nvccli` in the same manner as the legacy `--nv` binding approach. Pull the large container to a SIF file:

```
$ singularity pull docker://tensorflow/tensorflow:latest-gpu
...
INFO:    Creating SIF file...
INFO:    Build complete: tensorflow_latest-gpu.sif
```

Then run the container with `nvidia-container-cli` GPU support:

```
$ singularity run --nv --nvccli tensorflow_latest-gpu.sif
INFO:    Setting --writable-tmpfs (required by nvidia-container-cli)

________                            ______________
___  __/_____________________________  ____/__  /________      __
__  /  _  _ \_  __ \_  ___/  __ \_  ___/_  /_   __  /_  __ \_ | /| / /
_  /   /  __/  / / /(__  )/ /_/ /  /   _  __/   _  / / /_/ /_ |/ |/ /
/_/    \___//_/ /_//____/ \____//_/    /_/      /_/  \____/____/|__/
```

(continues on next page)

(continued from previous page)

```
You are running this container as user with ID 1000 and group 1000,
which should map to the ID and group for your user on the Docker host. Great!

Singularity>
```

Note that `--writable--tmpfs` was automatically set, which allows files to be written inside the container to an ephemeral overlay that will be discarded on exit. This is required for the `nvidia-container-cli` functionality.

You can verify the GPU is available within the container by using the tensorflow `list_local_devices()` function:

```
Singularity> python
Python 2.7.15+ (default, Jul  9 2019, 16:51:35)
[GCC 7.4.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from tensorflow.python.client import device_lib
>>> print(device_lib.list_local_devices())
...
device_type: "GPU"
memory_limit: 14474280960
locality {
  bus_id: 1
  links {
  }
}
incarnation: 13349913758992036690
physical_device_desc: "device: 0, name: Tesla T4, pci bus id: 0000:00:1e.0, compute␣
↪capability: 7.5"
...
```

GPU Selection

When running with `--nvccli`, by default SingularityCE will expose all GPUs on the host inside the container. This mirrors the functionality of the legacy GPU support for the most common use-case.

Setting the `SINGULARITY_CUDA_VISIBLE_DEVICES` environment variable before running a container is still supported, to control which GPUs are used by CUDA programs that honor `CUDA_VISIBLE_DEVICES`. However, more powerful GPU isolation is possible using the `--contain` flag and `NVIDIA_VISIBLE_DEVICES` environment variable. This controls which GPU devices are bound into the `/dev` tree in the container.

For example, to pass only the 2nd and 3rd GPU into a container running on a system with 4 GPUs, run the following:

```
$ export NVIDIA_VISIBLE_DEVICES=1,2
$ singularity run --contain --nv --nvccli mycontainer.sif
```

Note that:

- `NVIDIA_VISIBLE_DEVICES` is not prepended with `SINGULARITY_` as this variable controls container setup, and is not passed into the container.
- The GPU device identifiers start at 0, so 1,2 refers to the 2nd and 3rd GPU.
- You can use GPU UUIDs in place of numeric identifiers. Use `nvidia-smi -L` to list both numeric IDs and UUIDs available on the system.

- `all` can be used to pass all available GPUs into the container.

If you use `--contain` without setting `NVIDIA_VISIBLE_DEVICES`, no GPUs will be available in the container, and a warning will be shown:

```
$ singularity run --contain --nv --nvccli mycontainer.sif
WARNING: When using nvidia-container-cli with --contain NVIDIA_VISIBLE_DEVICES
must be set or no GPUs will be available in container.
```

To restore the behaviour of the legacy GPU handling, set `NVIDIA_VISIBLE_DEVICES=0` when running with `--contain`.

If your system contains Ampere or newer GPUs that support virtual MIG devices, you can specify MIG identifiers / UUIDs.

```
$ export NVIDIA_VISIBLE_DEVICES=MIG-GPU-5c89852c-d268-c3f3-1b07-005d5ae1dc3f/7/0
```

SingularityCE does not configure MIG partitions. It is expected that these would be statically configured by the system administrator, or setup dynamically by a job scheduler / workflow system according to the requirements of the job.

Other GPU Options

In `--nvccli` mode, SingularityCE understands the following additional environment variables. Note that these environment variables are read from the environment where `singularity` is run. SingularityCE does not currently read these settings from the container environment.

- `NVIDIA_DRIVER_CAPABILITIES` controls which libraries and utilities are mounted in the container, to support different requirements. The default value under SingularityCE is `compute,utility`, which will provide CUDA functionality and basic utilities such as `nvidia-smi`. Other options include `graphics` for OpenGL/Vulkan support, `video` for the codecs SDK, and `display` to use X11 from a container.
- `NVIDIA_REQUIRE_*` variables allow specifying requirements, which will be checked by `nvidia-container-cli` prior to starting the container. Constraints can be set on `cuda`, `driver`, `arch`, and `brand` values. Docker/OCI images may set these variables inside the container, to indicate runtime requirements. However, these container variables are not yet interpreted by SingularityCE.
- `NVIDIA_DISABLE_REQUIRE` will disable the enforcement of any `NVIDIA_REQUIRE_*` requirements that are set.

Full details of the supported values for these environment variables can be found in the container-toolkit guide:

https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/user-guide.html#environment-variables-oci-spec

6.4.3 AMD GPUs & ROCm

SingularityCE 3.5 adds a `--rocm` flag to support GPU compute with the ROCm framework using AMD Radeon GPU cards.

Commands that `run`, or otherwise execute containers (`shell`, `exec`) can take an `--rocm` option, which will setup the container's environment to use a Radeon GPU and the basic ROCm libraries to run a ROCm enabled application. The `--rocm` flag will:

- Ensure that the `/dev/dri/` device entries are available inside the container, so that the GPU cards in the host are accessible.
- Locate and bind the basic ROCm libraries from the host into the container, so that they are available to the container, and match the kernel GPU driver on the host.

- Set the `LD_LIBRARY_PATH` inside the container so that the bound-in version of the ROCm libraries are used by application run inside the container.

Requirements

To use the `--rocm` flag to run a CUDA application inside a container you must ensure that:

- The host has a working installation of the `amdgpu` driver, and a compatible version of the basic ROCm libraries. The host *does not* need to have an X server running, unless you want to run graphical apps from the container.
- The ROCm libraries are in the system's library search path.
- The application inside your container was compiled for a ROCm version that is compatible with the ROCm version on your host.

These requirements can be satisfied by following the requirements on the ROCm web site

Example - tensorflow-rocm

Tensorflow is commonly used for machine learning projects, but can be difficult to install on older systems, and is updated frequently. Running tensorflow from a container removes installation problems and makes trying out new versions easy.

The rocm tensorflow repository on Docker Hub contains Radeon GPU supporting containers, that will use ROCm for processing. You can view the available versions on the tags page on Docker Hub

The container is large, so it's best to build or pull the docker image to a SIF before you start working with it:

```
$ singularity pull docker://rocm/tensorflow:latest
...
INFO:    Creating SIF file...
INFO:    Build complete: tensorflow_latest.sif
```

Then run the container with GPU support:

```
$ singularity run --rocm tensorflow_latest.sif
```

You can verify the GPU is available within the container by using the tensorflow `list_local_devices()` function:

```
Singularity> ipython
Python 3.5.2 (default, Jul 10 2019, 11:58:48)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.8.0 -- An enhanced Interactive Python. Type '?' for help.
>>> from tensorflow.python.client import device_lib
...
>>> print(device_lib.list_local_devices())
...
2019-11-14 16:33:42.750509: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1651]␣
↪Found device 0 with properties:
name: Lexa PRO [Radeon RX 550/550X]
AMDGPU ISA: gfx803
memoryClockRate (GHz) 1.183
pciBusID 0000:09:00.0
...
```

6.4.4 OpenCL Applications

Both the `--rocm` and `--nv` flags will bind the vendor OpenCL implementation libraries into a container that is being run. However, these libraries will not be used by OpenCL applications unless a vendor icd file is available under `/etc/OpenCL/vendors` that directs OpenCL to use the vendor library.

The simplest way to use OpenCL in a container is to `--bind /etc/OpenCL` so that the icd files from the host (which match the bound-in libraries) are present in the container.

Example - Blender OpenCL

The Sylabs examples repository contains an example container definition for the 3D modelling application 'Blender'.

The latest versions of Blender supports OpenCL rendering. You can run Blender as a graphical application that will make use of a local Radeon GPU for OpenCL compute using the container that has been published to the Sylabs library:

```
$ singularity exec --rocm --bind /etc/OpenCL library://sylabs/examples/blender blender
```

Note the *exec* used as the *runscript* for this container is setup for batch rendering (which can also use OpenCL).

CHAPTER

SEVEN

GET INVOLVED

We'd love you to get involved in the SingularityCE community! Whether through contributing feature and fixes, helping to answer questions from other users, or simply testing new releases.

7.1 Contributing

SingularityCE is an open source project, meaning we have the challenge of limited resources. We are grateful for any support that you can offer. Helping other users, raising issues, helping write documentation, or contributing code are all ways to help!

7.1.1 Join the community

This is a huge endeavor, and your help would be greatly appreciated! Post to online communities about SingularityCE, and request that your distribution vendor, service provider, and system administrators include SingularityCE for you!

SingularityCE Google Group

If you have been using SingularityCE and having good luck with it, join our Google Group and help out other users!

SingularityCE on Slack

Many of our users come to Slack for quick help with an issue. You can find us at singularityce.slack.com.

7.1.2 Raise an Issue

For general bugs/issues, you can open an issue at the GitHub repo. However, if you find a security related issue/problem, please email Sylabs directly at security@sylabs.io. More information about the Sylabs security policies and procedures can be found here

7.1.3 Write Documentation

We (like almost all open source software providers) have a documentation dilemma... We tend to focus on the code features and functionality before working on documentation. And there is very good reason for this: we want to share the love so nobody feels left out!

You can contribute to the documentation by raising an issue to suggest an improvement or by sending a pull request on our repository for documentation.

The current documentation is generated with:

- reStructured Text (RST) and ReadTheDocs.

Other dependencies include:

- Python 3.5 or newer.
- Sphinx.

More information about contributing to the documentation, instructions on how to install the dependencies, and how to generate the files can be obtained here.

For more information on using Git and GitHub to create a pull request suggesting additions and edits to the docs, see the *section on contributing to the code*. The procedure is identical for contributions to the documentation and the code base.

7.1.4 Contribute to the code

We use the traditional GitHub Flow to develop. This means that you fork the main repo, create a new branch to make changes, and submit a pull request (PR) to the main branch.

Check out our official CONTRIBUTING.md document, which also includes a code of conduct.

Step 1. Fork the repo

To contribute to SingularityCE, you should obtain a GitHub account and fork the SingularityCE repository. Once forked, clone your fork of the repo to your computer. (Obviously, you should replace `your-username` with your GitHub username.)

```
$ git clone https://github.com/your-username/singularity.git && \
    cd singularity/
```

Step 2. Checkout a new branch

Branches are a way of isolating your features from the main branch. Given that we've just cloned the repo, we will probably want to make a new branch from main in which to work on our new feature. Lets call that branch `new-feature`:

```
$ git checkout main && \
    git checkout -b new-feature
```

Note: You can always check which branch you are in by running `git branch`.

Step 3. Make your changes

On your new branch, go nuts! Make changes, test them, and when you are happy commit the changes to the branch:

```
$ git add file-changed1 file-changed2...

$ git commit -m "what changed?"
```

This commit message is important - it should describe exactly the changes that you have made. Good commit messages read like so:

```
$ git commit -m "changed function getConfig in functions.go to output csv to fix #2"

$ git commit -m "updated docs about shell to close #10"
```

The tags `close #10` and `fix #2` are referencing issues that are posted on the upstream repo where you will direct your pull request. When your PR is merged into the main branch, these messages will automatically close the issues, and further, they will link your commits directly to the issues they intend to fix. This will help future maintainers understand your contribution, or (hopefully not) revert the code back to a previous version if necessary.

Step 4. Push your branch to your fork

When you are done with your commits, you should push your branch to your fork (and you can also continuously push commits here as you work):

```
$ git push origin new-feature
```

Note that you should always check the status of your branches to see what has been pushed (or not):

```
$ git status
```

Step 5. Submit a Pull Request

Once you have pushed your branch, then you can go to your fork (in the web GUI on GitHub) and submit a Pull Request. Regardless of the name of your branch, your PR should be submitted to the Sylabs `main` branch. Submitting your PR will open a conversation thread for the maintainers of SingularityCE to discuss your contribution. At this time, the continuous integration that is linked with the code base will also be executed. If there is an issue, or if the maintainers suggest changes, you can continue to push commits to your branch and they will update the Pull Request.

Step 6. Keep your branch in sync

Cloning the repo will create an exact copy of the SingularityCE repository at that moment. As you work, your branch may become out of date as others merge changes into the upstream main. In the event that you need to update a branch, you will need to follow the next steps:

```
$ git remote add upstream https://github.com/sylabs/singularity.git && # to add a new␣
↪remote named "upstream" \
    git checkout main && # or another branch to be updated \
    git pull upstream main && \
    git push origin main && # to update your fork \
    git checkout new-feature && \
    git merge main
```

CHAPTER

EIGHT

REFERENCE

8.1 Appendix

8.1.1 SingularityCE's environment variables

SingularityCE 3.0 comes with some environment variables you can set or modify depending on your needs. You can see them listed alphabetically below with their respective functionality.

A

1. **SINGULARITY_ADD_CAPS**: To specify a list (comma separated string) of capabilities to be added. Default is an empty string.
2. **SINGULARITY_ALL**: List all the users and groups capabilities.
3. **SINGULARITY_ALLOW_SETUID**: To specify that setuid binaries should or not be allowed in the container. (root only) Default is set to false.
4. **SINGULARITY_ALLOW_UNSIGNED**: Set to true to allow pushing unsigned SIF images to a `library://` destination. Default is false.
5. **SINGULARITY_APP** and **SINGULARITY_APPNAME**: Sets the name of an application to be run inside a container.
6. **SINGULARITY_APPLY_CGROUPS**: Used to apply cgroups from an input file for container processes. (it requires root privileges)
7. **SINGULARITY_ARCH** and **SINGULARITY_PULL_ARCH**: Set the architecture (e.g. `arm64`) of an image to pull from a `library://` or OCI source. Defaults to the host architecture.

B

1. **SINGULARITY_BINDPATH** and **SINGULARITY_BIND**: Comma separated string `source:<dest>` list of paths to bind between the host and the container.
2. **SINGULARITY_BLKIO_WEIGHT**: Specify a relative weight for block device access during contention. Range 10-1000. Default is 0 (disabled).
3. **SINGULARITY_BLKIO_WEIGHT_DEVICE**: Specify a relative weight for block device access during contention on a specific device. Must be supplied in `<device path>:weight` format. Default is unset.
4. **SINGULARITY_BOOT**: Set to false by default, considers if executing `/sbin/init` when container boots (root only).

5. **SINGULARITY_BUILD_ARCH**: Specify an architecture to use when building a container via the remote build service (*–remote*).
6. **SINGULARITY_BUILDER**: To specify the remote builder service URL. Defaults to our remote builder.

C

1. **SINGULARITY_CACHEDIR**: Specifies the directory for image downloads to be cached in. See *Cache Folders*.
2. **SINGULARITY_CAP_GROUP**: Specify a group to modify when managing permitted capabilities with the `capability` command.
3. **SINGULARITY_CAP_USER**: Specify a user to modify when managing permitted capabilities with the `capability` command.
4. **SINGULARITY_CLEANENV**: Specifies if the environment should be cleaned or not before running the container. Default is set to false.
5. **SINGULARITY_COMPAT**: Set to true to enable Docker/OCI compatibility mode. Equivalent to setting `--containall --no-eval --no-init --no-umask --writable-tmpfs`. Default is false for the native runtime, true in OCI-mode.
6. **SINGULARITY_CONFIG_FILE**: Use a custom `singularity.conf` configuration file. Only supported for non-root users in non-setuid mode.
7. **SINGULARITY_CONTAIN**: To use minimal `/dev` and empty other directories (e.g. `/tmp` and `$HOME`) instead of sharing filesystems from your host. Default is set to false.
8. **SINGULARITY_CONTAINALL**: To contain not only file systems, but also PID, IPC, and environment. Default is set to false.
9. **SINGULARITY_CONTAINLIBS**: Used to specify a string of file names (comma separated string) to bind to the `/.singularity.d/libs` directory.
10. **SINGULARITY_CPU_SHARES**: Specify a relative share of CPU time available to the container. Default is -1 (disabled).
11. **SINGULARITY_CPUS**: Specify a fractional number of CPUs available to the container. Default is unset.
12. **SINGULARITY_CPUSET_CPUS**: Specify a list or range of CPU cores available to the container. Default is unset.
13. **SINGULARITY_CPUSET_MEMS**: Specify a list or range of memory nodes available to the container. Default is unset.
14. **SINGULARITY_CWD** (deprecated **SINGULARITY_PWD** and **SINGULARITY_TARGET_PWD**): The initial working directory for payload process inside the container.

D

1. **SINGULARITY_DEBUG**: Enable debug output when set. Equivalent to `-d / --debug`.
2. **SINGULARITY_DEFFILE**: Shows the SingularityCE recipe that was used to generate the image.
3. **SINGULARITY_DESC**: Contains a description of the capabilities.
4. **SINGULARITY_DETACHED**: To submit a build job and print the build ID (no real-time logs and also requires `--remote`). Default is set to false.
5. **SINGULARITY_DISABLE_CACHE**: To disable all caching of docker/oci, library, oras, etc. downloads and built SIFs. Default is set to false.

6. **SINGULARITY_DNS**: A list of the DNS server addresses separated by commas to be added in `resolv.conf`.
7. **SINGULARITY_DOCKER_HOST**: Host address / socket to use when pulling images from a `docker-daemon` source. `DOCKER_HOST` without the `SINGULARITY_` prefix is also accepted.
8. **SINGULARITY_DOCKER_LOGIN**: To specify the interactive prompt for docker authentication.
9. **SINGULARITY_DOCKER_PASSWORD**: To specify the password for docker authentication. `DOCKER_PASSWORD` without the `SINGULARITY_` prefix is also accepted.
10. **SINGULARITY_DOCKER_USERNAME**: To specify the username for docker authentication. `DOCKER_USERNAME` without the `SINGULARITY_` prefix is also accepted.
11. **SINGULARITY_DOWNLOAD_CONCURRENCY**: To specify how many concurrent streams when downloading (pulling) an image from cloud library.
12. **SINGULARITY_DOWNLOAD_PART_SIZE**: To specify the size of each part (bytes) when concurrent downloads are enabled.
13. **SINGULARITY_DOWNLOAD_BUFFER_SIZE**: To specify the transfer buffer size (bytes) when concurrent downloads are enabled.
14. **SINGULARITY_DROP_CAPS**: To specify a list (comma separated string) of capabilities to be dropped. Default is an empty string.

E

1. **SINGULARITY_ENCRYPTION_PASSPHRASE**: Used to specify the plaintext passphrase to encrypt the container.
2. **SINGULARITY_ENCRYPTION_PEM_PATH**: Used to specify the path of the file containing public or private key to encrypt the container in PEM format.
3. **SINGULARITY_ENV_FILE**: Specify a file containing `KEY=VAL` environment variables that should be set in the container.
4. **SINGULARITY_ENVIRONMENT**: Set during a build to the path to a file into which `KEY=VAL` environment variables can be added. The file is evaluated at container startup.
5. **SINGULARITYENV_***: Allows you to transpose variables into the container at runtime. You can see more in detail how to use this variable in our *environment and metadata section*.
6. **SINGULARITYENV_APPEND_PATH**: Used to append directories to the end of the `$PATH` environment variable. You can see more in detail on how to use this variable in our *environment and metadata section*.
7. **SINGULARITYENV_PATH**: A specified path to override the `$PATH` environment variable within the container. You can see more in detail on how to use this variable in our *environment and metadata section*.
8. **SINGULARITYENV_PREPEND_PATH**: Used to prepend directories to the beginning of `$PATH` environment variable. You can see more in detail on how to use this variable in our *environment and metadata section*.

F

1. **SINGULARITY_FAKEROOT**: Run or build a container using a user namespace with a root uid/gid mapping.
2. **SINGULARITY_FIXPERMS**: Set to true to ensure owner has `rwX` permissions on all files in a container built from an OCI source.
3. **SINGULARITY_FORCE**: Skip confirmation for destructive actions, e.g. overwriting a container image or killing an instance.
4. **SINGULARITY_FUSESPEC**: A FUSE filesystem mount specification of the form '<type>:<fuse command> <mountpoint>', that will be mounted in the container.

H

1. **SINGULARITY_HELPFILE**: Specifies the runscript helpfile, if it exists.
2. **SINGULARITY_HOME** : A home directory specification, it could be a source or destination path. The source path is the home directory outside the container and the destination overrides the home directory within the container.
3. **SINGULARITY_HOSTNAME**: The container's hostname.

I

1. **SINGULARITY_IMAGE**: Filename of the container.

J

1. **SINGULARITY_JSON**: Use JSON as an input or output format. Applies to the `build` and `instance list` commands. Default is false.

K

1. **SINGULARITY_KEEP_PRIVS**: To let root user keep privileges in the container. Default is set to false.

L

1. **SINGULARITY_LABELS**: Specifies the labels associated with the image.
2. **SINGULARITY_LIBRARY**: Specifies the library to pull from. Default is set to our Cloud Library.
3. **SINGULARITY_LOCAL_VERIFY**: Set to true to only use the local keyring when verifying PGP signed SIF images. Disables retrieval of public keys from configured keyservers. Default is false.
4. **SINGULARITY_LOGIN_USERNAME**: Set the username to use when logging in to a remote endpoint, registry, or keyserver.
5. **SINGULARITY_LOGIN_PASSWORD**: Set the password to use when logging in to a remote endpoint, registry, or keyserver.
6. **SINGULARITY_LOGIN_INSECURE**: Set to true to use HTTP (not HTTPS) when logging in to a remote endpoint. Default is false.
7. **SINGULARITY_LOGS**: Set to true to show the path to instance log files in `instance list` output. Default is false.

M

1. **SINGULARITY_MEMORY**: Specify a memory limit in bytes for the container. Default is unset (no limit).
2. **SINGULARITY_MEMORY_RESERVATION**: Specify a memory soft limit in bytes for the container. Default is unset (no limit).
3. **SINGULARITY_MEMORY_SWAP**: Specify a limit for memory + swap usage by the container. Default is unset. Effect depends on **SINGULARITY_MEMORY**.
4. **SINGULARITY_MOUNT**: To specify host to container mounts, using the syntax understood by the `--mount` flag. Multiple mounts should be separated by newline characters.

N

1. **SINGULARITY_NAME**: Specifies a custom image name.
2. **SINGULARITY_NETWORK**: Used to specify a desired network. If more than one parameters is used, addresses should be separated by commas, where each network will bring up a dedicated interface inside the container.
3. **SINGULARITY_NETWORK_ARGS**: To specify the network arguments to pass to CNI plugins.
4. **SINGULARITY_NOCLEANUP**: To not clean up the bundle after a failed build, this can be helpful for debugging. Default is set to false.
5. **SINGULARITY_NO_COMPAT**: Set to true to emulate traditional Singularity behavior (e.g. home, cwd mounts) when running in OCI mode.
6. **SINGULARITY_NO_HTTPS** and **SINGULARITY_NOHTTPS**: Set to true to use HTTP (not HTTPS) to communicate with registry servers. Default is false.
7. **SINGULARITY_NO_EVAL**: Set to true in order to prevent SingularityCE performing shell evaluation on environment variables / runscript arguments at startup.
8. **SINGULARITY_NO_HOME**: Considers not mounting users home directory if home is not the current working directory. Default is set to false.
9. **SINGULARITY_NO_INIT** and **SINGULARITY_NOSHIMINIT**: Considers not starting the `shim` process with `--pid`.
10. **SINGULARITY_NO_MOUNT**: Disable an automatic mount that has been set in `singularity.conf`. Accepts `proc` / `sys` / `dev` / `devpts` / `home` / `tmp` / `hostfs` / `cwd`, or the source path for a system specifc bind.
11. **SINGULARITY_NO_NV**: Flag to disable NVIDIA support. Opposite of `SINGULARITY_NV`.
12. **SINGULARITY_NO_PID**: Set to true to disable the PID namespace, when it is inferred by other options (e.g.``–containall``)
13. **SINGULARITY_NO_PRIVS**: To drop all the privileges from root user in the container. Default is false.
14. **SINGULARITY_NO_SETGROUPS**: When set to true, do not clear supplementary group membership when entering a fakeroot user namespace. Default is false.
15. **SINGULARITY_NOTEST**: Set to true to disable execution of `%test` sections when building a container.
16. **SINGULARITY_NO_UMASK**: Set to true to prevent host umask propagating to container, and use a default 0022 unmask instead. Default is false.
17. **SINGULARITY_NV**: To enable NVIDIA GPU support. Default is set to false.
18. **SINGULARITY_NVCCLI**: To use nvidia-container-cli for container GPU setup (experimental).

19. **SINGULARITY_NO_TMP_SANDBOX**: Set to true to disable fall-back approach of extracting a container to a temporary sandbox when SIF / OCI-SIF mounts cannot be used. Default is false. Also configurable via `tmp sandbox` in `singularity.conf`.

O

1. **SINGULARITY_OCI**: Set to true to run containers in OCI mode, and pull OCI images to the OCI-SIF format. Default is taken from `oci mode` directive in `singularity.conf`.
2. **SINGULARITY_NO_OCI**: Set to true to disable OCI mode, and pull OCI images to the native SIF format, when `oci mode` is enabled in `singularity.conf`.
3. **SINGULARITY_OOM_KILL_DISABLE**: Set to true to disable OOM killer for container processes, if possible. Default is false.
4. **SINGULARITY_OVERLAY** and **SINGULARITY_OVERLAYIMAGE**: To indicate the use of an overlay file system image for persistent data storage or as read-only layer of container.

P

1. **SINGULARITY_PULLDIR** and **SINGULARITY_PULLFOLDER**: Specify destination directory when pulling a container image.
2. **SINGULARITY_PID_FILE**: When starting an instance, write the instance PID to the specified file.
3. **SINGULARITY_PIDS_LIMIT**: Specify maximum number of processes that the container may spawne. Default is 0 (no limit).
4. **SINGULARITY_PLATFORM**: Set the platform (e.g. `linux/arm/v7`) of an image to pull from a `library://` or OCI source. Defaults to the host platform. Note that `library://` pulls ignore the platform variant.

R

1. **SINGULARITY_REMOTE**: Set to true to build an image remotely using a remote build service. Default is set to false.
2. **SINGULARITY_ROOTFS**: During a build `SINGULARITY_ROOTFS` is set to the path of the rootfs for the container. It can be used within a definition file to manipulate the rootfs (e.g. from the `%setup` section).
3. **SINGULARITY_ROCM**: Set to true to expose ROCm devices and libraries inside the container. Default is false.
4. **SINGULARITY_RUNSCRIPT**: Specifies the runscript of the image.

S

1. **SINGULARITY_SANDBOX**: Set to true to specify that the format of the image should be a sandbox. Default is set to false.
2. **SINGULARITY_SCRATCH** and **SINGULARITY_SCRATCHDIR**: Used to include a scratch directory within the container that is linked to a temporary directory. (use -W to force location)
3. **SINGULARITY_SECTION**: Set to specify a comma separated string of all the sections to be run from the deffile (setup, post, files, environment, test, labels, none)
4. **SINGULARITY_SECURITY**: Used to enable security features. (SELinux, Apparmor, Seccomp)
5. **SINGULARITY_SECRET**: Lists all the private keys instead of the default which display the public ones.

6. **SINGULARITY_SHELL**: The path to the program to be used as an interactive shell.
7. **SINGULARITY_SIF_FUSE**: (experimental) Set to true to attempt to mount SIF images with `squashfuse` in unprivileged user namespace workflows.
8. **SINGULARITY_SIGNAL**: Specifies the signal to send to an instance with `singularity instance stop`.
9. **SINGULARITY_SIGN_KEY**: Set the path to a key file to be used when signing a SIF image.
10. **SINGULARITY_SPARSE**: Set to true to create sparse overlay image files with the overlay command.

T

1. **SINGULARITY_TEST**: Specifies the test script for the image.
2. **SINGULARITY_TMPDIR**: Specify a location for temporary files to be used when pulling and building container images. See *Temporary Folders*.

U

1. **SINGULARITY_UNSHARE_PID**: To specify that the container will run in a new PID namespace. Default is set to false.
2. **SINGULARITY_UNSHARE_IPC**: To specify that the container will run in a new IPC namespace. Default is set to false.
3. **SINGULARITY_UNSHARE_NET**: To specify that the container will run in a new network namespace (sets up a bridge network interface by default). Default is set to false.
4. **SINGULARITY_UNSHARE_UTS**: To specify that the container will run in a new UTS namespace. Default is set to false.
5. **SINGULARITY_UPDATE**: To run the definition over an existing container (skips the header). Default is set to false.
6. **SINGULARITY_URL**: Specifies the key server `URL`.
7. **SINGULARITY_USER**: As root, specify a user to manage that user's instances with the `instance` commands.
8. **SINGULARITY_USERNS** and **SINGULARITY_UNSHARE_USERNS**: To specify that the container will run in a new user namespace, allowing SingularityCE to run completely unprivileged on recent kernels. This may not support every feature of SingularityCE. (Sandbox image only). Default is set to false.

V

1. **SINGULARITY_VERIFY_CERTIFICATE**: Set the path to a PEM file containing the certificate to be used when verifying an x509 signed SIF image.
2. **SINGULARITY_VERIFY_INTERMEDIATES**: Set the path to a PEM file containing an intermediate certificate / chain to be used when verifying an x509 signed SIF image.
3. **SINGULARITY_VERIFY_KEY**: Set the path to a key file to be used when verifying a key signed SIF image.
4. **SINGULARITY_VERIFY_OCSP**: Set to true to enable OCSP verification of certificates. Default is false.
5. **SINGULARITY_VERIFY_ROOTS**: Set the path to a PEM file containing root certificate(s) to be used when verifying an x509 signed SIF image.

W

1. **SINGULARITY_WORKDIR**: The working directory to be used for `/tmp`, `/var/tmp` and `$HOME` (if `-c` or `--contain` was also used)
2. **SINGULARITY_WRITABLE**: By default, all SingularityCE containers are available as read only, this option makes the file system accessible as read/write. Default set to false.
3. **SINGULARITY_WRITABLE_TMPFS**: Makes the file system accessible as read-write with non-persistent data (with overlay support only). Default is set to false.

8.1.2 Build Modules

`library` bootstrap agent

Overview

You can use an existing container on the Container Library as your "base," and then add customization. This allows you to build multiple images from the same starting point. For example, you may want to build several containers with the same custom python installation, the same custom compiler toolchain, or the same base MPI installation. Instead of building these from scratch each time, you could create a base container on the Container Library and then build new containers from that existing base container adding customizations in `%post`, `%environment`, `%runscript`, etc.

Keywords

```
Bootstrap: library
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
From: <entity>/<collection>/<container>:<tag>
```

The `From` keyword is mandatory. It specifies the container to use as a base. `entity` is optional and defaults to `library`. `collection` is optional and defaults to `default`. This is the correct namespace to use for some official containers (`alpine` for example). `tag` is also optional and will default to `latest`.

```
Library: http://custom/library
```

The Library keyword is optional. It will default to `https://library.sylabs.io`.

```
Fingerprints: 22045C8C0B1004D058DE4BEDA20C27EE7FF7BA84
```

The Fingerprints keyword is optional. It specifies one or more comma separated fingerprints corresponding to PGP public keys. If present, the bootstrap image will be verified and the build will only proceed if it is signed by keys matching *all* of the specified fingerprints.

`docker` bootstrap agent

Overview

Docker images are comprised of layers that are assembled at runtime to create an image. You can use Docker layers to create a base image, and then add your own custom software. For example, you might use Docker's Ubuntu image layers to create an Ubuntu SingularityCE container. You could do the same with CentOS, Debian, Arch, Suse, Alpine, BusyBox, etc.

Or maybe you want a container that already has software installed. For instance, maybe you want to build a container that uses CUDA and cuDNN to leverage the GPU, but you don't want to install from scratch. You can start with one of the `nvidia/cuda` containers and install your software on top of that.

Or perhaps you have already invested in Docker and created your own Docker containers. If so, you can seamlessly convert them to SingularityCE with the `docker` bootstrap module.

Keywords

```
Bootstrap: docker
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
From: <registry>/<namespace>/<container>:<tag>@<digest>
```

The `From` keyword is mandatory. It specifies the container to use as a base. `registry` is optional and defaults to `index.docker.io`. `namespace` is optional and defaults to `library`. This is the correct namespace to use for some official containers (ubuntu for example). `tag` is also optional and will default to `latest`

See *SingularityCE and Docker* for more detailed info on using Docker registries.

```
Registry: http://custom_registry
```

The Registry keyword is optional. It will default to `index.docker.io`.

```
Namespace: namespace
```

The Namespace keyword is optional. It will default to `library`.

Notes

Docker containers are stored as a collection of tarballs called layers. When building from a Docker container the layers must be downloaded and then assembled in the proper order to produce a viable file system. Then the file system must be converted to Singularity Image File (sif) format.

Building from Docker Hub is not considered reproducible because if any of the layers of the image are changed, the container will change. If reproducibility is important to your workflow, consider hosting a base container on the Container Library and building from it instead.

For detailed information about setting your build environment see *Build Customization*.

shub bootstrap agent

Overview

You can use an existing container on Singularity Hub as your "base," and then add customization. This allows you to build multiple images from the same starting point. For example, you may want to build several containers with the same custom python installation, the same custom compiler toolchain, or the same base MPI installation. Instead of building these from scratch each time, you could create a base container on Singularity Hub and then build new containers from that existing base container adding customizations in %post , %environment, %runscript, etc.

Keywords

```
Bootstrap: shub
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
From: shub://<registry>/<username>/<container-name>:<tag>@digest
```

The From keyword is mandatory. It specifies the container to use as a base. registry is optional and defaults to ``singularity-hub.org. tag and digest are also optional. tag defaults to latest and digest can be left blank if you want the latest build.

Notes

When bootstrapping from a Singularity Hub image, all previous definition files that led to the creation of the current image will be stored in a directory within the container called /.singularity.d/bootstrap_history. SingularityCE will also alert you if environment variables have been changed between the base image and the new image during bootstrap.

oras bootstrap agent

Overview

Using, this module, a container from supporting OCI Registries - Eg: ACR (Azure Container Registry), local container registries, etc can be used as your "base" image and later customized. This allows you to build multiple images from the same starting point. For example, you may want to build several containers with the same custom python installation, the same custom compiler toolchain, or the same base MPI installation. Instead of building these from scratch each time, you could make use of oras to pull an appropriate base container and then build new containers by adding customizations in %post , %environment, %runscript, etc.

Keywords

```
Bootstrap: oras
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
From: registry/namespace/image:tag
```

The From keyword is mandatory. It specifies the container to use as a base. Also,``tag`` is mandatory that refers to the version of image you want to use.

`localimage` bootstrap agent

This module allows you to build a container from an existing SingularityCE container on your host system. The name is somewhat misleading because your container can be in either image or directory format.

Overview

You can use an existing container image as your "base", and then add customization. This allows you to build multiple images from the same starting point. For example, you may want to build several containers with the same custom python installation, the same custom compiler toolchain, or the same base MPI installation. Instead of building these from scratch each time, you could start with the appropriate local base container and then customize the new container in `%post`, `%environment`, `%runscript`, etc.

Keywords

```
Bootstrap: localimage
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
From: /path/to/container/file/or/directory
```

The `From` keyword is mandatory. It specifies the local container to use as a base.

```
Fingerprints: 22045C8C0B1004D058DE4BEDA20C27EE7FF7BA84
```

The Fingerprints keyword is optional. It specifies one or more comma separated fingerprints corresponding to PGP public keys. If present, and the `From:` keyword points to a SIF format image, it will be verified and the build will only proceed if it is signed by keys matching *all* of the specified fingerprints.

Notes

When building from a local container, all previous definition files that led to the creation of the current container will be stored in a directory within the container called `/.singularity.d/bootstrap_history`. SingularityCE will also alert you if environment variables have been changed between the base image and the new image during bootstrap.

`yum` bootstrap agent

This module allows you to build a Red Hat/CentOS/Scientific Linux style container from a mirror URI.

Overview

Use the `yum` module to specify a base for a CentOS-like container. You must also specify the URI for the mirror you would like to use.

Keywords

```
Bootstrap: yum
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
OSVersion: 7
```

The OSVersion keyword is optional. It specifies the OS version you would like to use. It is only required if you have specified a %{OSVERSION} variable in the `MirrorURL` keyword.

```
MirrorURL: http://mirror.centos.org/centos-%{OSVERSION}/%{OSVERSION}/os/$basearch/
```

The MirrorURL keyword is mandatory. It specifies the URI to use as a mirror to download the OS. If you define the `OSVersion` keyword, then you can use it in the URI as in the example above.

```
Include: yum
```

The Include keyword is optional. It allows you to install additional packages into the core operating system. It is a best practice to supply only the bare essentials such that the `%post` section has what it needs to properly complete the build. One common package you may want to install when using the `yum` build module is YUM itself.

Notes

There is a major limitation with using YUM to bootstrap a container. The RPM database that exists within the container will be created using the RPM library and Berkeley DB implementation that exists on the host system. If the RPM implementation inside the container is not compatible with the RPM database that was used to create the container, RPM and YUM commands inside the container may fail. This issue can be easily demonstrated by bootstrapping an older RHEL compatible image by a newer one (e.g. bootstrap a Centos 5 or 6 container from a Centos 7 host).

In order to use the `yum` build module, you must have `yum` installed on your system. It may seem counter-intuitive to install YUM on a system that uses a different package manager, but you can do so. For instance, on Ubuntu you can install it like so:

```
$ sudo apt-get update && sudo apt-get install yum
```

`debootstrap` build agent

This module allows you to build a Debian/Ubuntu style container from a mirror URI.

Overview

Use the `debootstrap` module to specify a base for a Debian-like container. You must also specify the OS version and a URI for the mirror you would like to use.

Keywords

```
Bootstrap: debootstrap
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
OSVersion: xenial
```

The OSVersion keyword is mandatory. It specifies the OS version you would like to use. For Ubuntu you can use code words like `trusty` (14.04), `xenial` (16.04), and `yakkety` (17.04). For Debian you can use values like `stable`, `oldstable`, `testing`, and `unstable` or code words like `wheezy` (7), `jesse` (8), and `stretch` (9).

```
MirrorURL:  http://us.archive.ubuntu.com/ubuntu/
```

The MirrorURL keyword is mandatory. It specifies a URI to use as a mirror when downloading the OS.

```
Include: somepackage
```

The Include keyword is optional. It allows you to install additional packages into the core operating system. It is a best practice to supply only the bare essentials such that the `%post` section has what it needs to properly complete the build.

Notes

In order to use the `debootstrap` build module, you must have `debootstrap` installed on your system. On Ubuntu you can install it like so:

```
$ sudo apt-get update && sudo apt-get install debootstrap
```

On CentOS you can install it from the epel repos like so:

```
$ sudo yum update && sudo yum install epel-release && sudo yum install debootstrap.noarch
```

`arch` bootstrap agent

This module allows you to build a Arch Linux based container.

Overview

Use the `arch` module to specify a base for an Arch Linux based container. Arch Linux uses the aptly named `pacman` package manager (all puns intended).

Keywords

```
Bootstrap: arch
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

The Arch Linux bootstrap module does not name any additional keywords at this time. By defining the `arch` module, you have essentially given all of the information necessary for that particular bootstrap module to build a core operating system.

Notes

Arch Linux is, by design, a very stripped down, light-weight OS. You may need to perform a significant amount of configuration to get a usable OS. Please refer to this README.md and the Arch Linux example for more info.

`busybox` bootstrap agent

This module allows you to build a container based on BusyBox.

Overview

Use the `busybox` module to specify a BusyBox base for container. You must also specify a URI for the mirror you would like to use.

Keywords

```
Bootstrap: busybox
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
MirrorURL: https://www.busybox.net/downloads/binaries/1.26.1-defconfig-multiarch/busybox-
↪x86_64
```

The MirrorURL keyword is mandatory. It specifies a URI to use as a mirror when downloading the OS.

Notes

You can build a fully functional BusyBox container that only takes up ~600kB of disk space!

`zypper` bootstrap agent

This module allows you to build a Suse style container from a mirror URI.

Note: `zypper` version 1.11.20 or greater is required on the host system, as SingularityCE requires the `--releasever` flag.

Overview

Use the `zypper` module to specify a base for a Suse-like container. You must also specify a URI for the mirror you would like to use.

Keywords

```
Bootstrap: zypper
```

The Bootstrap keyword is always mandatory. It describes the bootstrap module to use.

```
OSVersion: 42.2
```

The OSVersion keyword is optional. It specifies the OS version you would like to use. It is only required if you have specified a %{OSVERSION} variable in the `MirrorURL` keyword.

```
Include: somepackage
```

The Include keyword is optional. It allows you to install additional packages into the core operating system. It is a best practice to supply only the bare essentials such that the `%post` section has what it needs to properly complete the build. One common package you may want to install when using the zypper build module is `zypper` itself.

`docker-daemon` bootstrap agent

Overview

`docker-daemon` allows you to build a SIF from any Docker image currently residing in the Docker daemon's internal storage:

```
$ docker images alpine
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
alpine              latest              965ea09ff2eb        7 weeks ago         5.55MB

$ singularity run docker-daemon:alpine:latest
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
Getting image source signatures
Copying blob 77cae8ab23bf done
Copying config 759e71f0d3 done
Writing manifest to image destination
Storing signatures
2019/12/11 14:53:24  info unpack layer:␣
↪sha256:eb7c47c7f0fd0054242f35366d166e6b041dfb0b89e5f93a82ad3a3206222502
INFO:    Creating SIF file...
Singularity>
```

The `SINGULARITY_DOCKER_HOST` or `DOCKER_HOST` environment variables may be set to instruct {SingularityCE} to pull images from a Docker daemon that is not running at the default location. For example, when using a virtualized Docker you may be instructed to set `DOCKER_HOST` e.g.

```
To connect the Docker client to the Docker daemon, please set
export DOCKER_HOST=tcp://192.168.59.103:2375
```

Keywords

In a definition file, the `docker-daemon` bootstrap agent requires the source container reference to be provided with the `From:` keyword:

```
Bootstrap: docker-daemon
From: <image>:<tag>
```

where both `<image>` and `<tag>` are mandatory fields that must be written explicitly.

`docker-archive` bootstrap agent

Overview

The `docker-archive` boostrap agent allows you to create a SingularityCE image from a docker image stored in a `docker save` formatted tar file:

```
$ docker save -o alpine.tar alpine:latest

$ singularity run docker-archive:$(pwd)/alpine.tar
INFO:    Converting OCI blobs to SIF format
INFO:    Starting build...
Getting image source signatures
Copying blob 77cae8ab23bf done
Copying config 759e71f0d3 done
Writing manifest to image destination
Storing signatures
2019/12/11 15:25:09  info unpack layer:␣
↪sha256:eb7c47c7f0fd0054242f35366d166e6b041dfb0b89e5f93a82ad3a3206222502
INFO:    Creating SIF file...
Singularity>
```

Keywords

In a definition file, the `docker-archive` bootstrap agent requires the path to the tar file containing the image to be specified with the `From:` keyword.

```
Bootstrap: docker-archive
From: <path-to-tar-file>
```

`scratch` bootstrap agent

The scratch bootstrap agent allows you to start from a completely empty container. You are then responsible for adding any and all executables, libraries etc. that are required. Starting with a scratch container can be useful when you are aiming to minimize container size, and have a simple application / static binaries.

Overview

A minimal container providing a shell can be created by copying the `busybox` static binary into an empty scratch container:

```
Bootstrap: scratch

%setup
    # Runs on host - fetch static busybox binary
    curl -o /tmp/busybox https://www.busybox.net/downloads/binaries/1.31.0-i686-uclibc/
↪busybox
    # It needs to be executable
    chmod +x /tmp/busybox

%files
    # Copy from host into empty container
    /tmp/busybox /bin/sh

%runscript
   /bin/sh
```

The resulting container provides a shell, and is 696KiB in size:

```
$ ls -lah scratch.sif
-rwxr-xr-x. 1 dave dave 696K May 28 13:29 scratch.sif

$ singularity run scratch.sif
WARNING: passwd file doesn't exist in container, not updating
WARNING: group file doesn't exist in container, not updating
Singularity> echo "Hello from a 696KiB container"
Hello from a 696KiB container
```

Keywords

```
Bootstrap: scratch
```

There are no additional keywords for the scratch bootstrap agent.

8.2 Command Line Interface

Below are links to the automatically generated CLI docs

8.2.1 singularity

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Synopsis

Singularity containers provide an application virtualization layer enabling mobility of compute via both application and environment portability. With Singularity one is capable of building a root file system that runs on any other Linux system where Singularity is installed.

```
singularity [global options...]
```

Examples

```
$ singularity help <command> [<subcommand>]
$ singularity help build
$ singularity help instance start
```

Options

```
-c, --config string   specify a configuration file (for root or unprivileged␣
↪installation only) (default "/usr/local/etc/singularity/singularity.conf")
-d, --debug           print debugging information (highest verbosity)
-h, --help            help for singularity
    --nocolor         print without color output (default False)
-q, --quiet           suppress normal output
-s, --silent          only print errors
-v, --verbose         print additional information
```

SEE ALSO

- *singularity build* - Build a Singularity image
- *singularity cache* - Manage the local cache
- *singularity capability* - Manage Linux capabilities for users and groups
- *singularity config* - Manage various singularity configuration (root user only)
- *singularity delete* - Deletes requested image from the library
- *singularity exec* - Run a command within a container
- *singularity inspect* - Show metadata for an image
- *singularity instance* - Manage containers running as services

- *singularity key* - Manage OpenPGP keys
- *singularity keyserver* - Manage singularity keyservers
- *singularity oci* - Manage OCI containers
- *singularity overlay* - Manage an EXT3 writable overlay image
- *singularity plugin* - Manage Singularity plugins
- *singularity pull* - Pull an image from a URI
- *singularity push* - Upload image to the provided URI
- *singularity registry* - Manage authentication to OCI/Docker registries
- *singularity remote* - Manage singularity remote endpoints
- *singularity run* - Run the user-defined default command within a container
- *singularity run-help* - Show the user-defined help for an image
- *singularity search* - Search a Container Library for images
- *singularity shell* - Run a shell within a container
- *singularity sif* - Manipulate Singularity Image Format (SIF) images
- *singularity sign* - Add digital signature(s) to an image
- *singularity test* - Run the user-defined tests within a container
- *singularity verify* - Verify digital signature(s) within an image
- *singularity version* - Show the version for Singularity

Auto generated by spf13/cobra on 23-Nov-2023

8.2.2 singularity build

Build a Singularity image

Synopsis

IMAGE PATH:

When Singularity builds the container, output can be one of a few formats:

default: The compressed Singularity read only image format (default) sandbox: This is a read-write container within a directory structure

note: It is a common workflow to use the "sandbox" mode for development of the container, and then build it as a default Singularity image for production use. The default format is immutable.

BUILD SPEC:

The build spec target is a definition (def) file, local image, or URI that can be used to create a Singularity container. Several different local target formats exist:

def file : This is a recipe for building a container (examples below) directory: A directory structure containing a (ch)root file system image: A local image on your machine (will convert to sif if

it is legacy format)

Targets can also be remote and defined by a URI of the following formats:

library:// an image library (default https://cloud.sylabs.io/library) docker:// a Docker/OCI registry (default Docker Hub) shub:// a Singularity registry (default Singularity Hub) oras:// an OCI registry that holds SIF files using ORAS

```
singularity build [local options...] <IMAGE PATH> <BUILD SPEC>
```

Examples

```
DEF FILE BASE OS:

    Library:
        Bootstrap: library
        From: debian:9

    Docker:
        Bootstrap: docker
        From: tensorflow/tensorflow:latest
        IncludeCmd: yes # Use the CMD as runscript instead of ENTRYPOINT

    Singularity Hub:
        Bootstrap: shub
        From: singularityhub/centos

    YUM/RHEL:
        Bootstrap: yum
        OSVersion: 7
        MirrorURL: http://mirror.centos.org/centos-%{OSVERSION}/%{OSVERSION}/os/x86_64/
        Include: yum

    Debian/Ubuntu:
        Bootstrap: debootstrap
        OSVersion: trusty
        MirrorURL: http://us.archive.ubuntu.com/ubuntu/

    Local Image:
        Bootstrap: localimage
        From: /home/dave/starter.img

    Scratch:
        Bootstrap: scratch # Populate the container with a minimal rootfs in %setup

DEFFILE SECTIONS:

The following sections are presented in the order of processing, with the exception
that labels and environment can also be manipulated in %post.

    %pre
        echo "This is a scriptlet that will be executed on the host, as root before"
        echo "the container has been bootstrapped. This section is not commonly used."
```

(continues on next page)

(continued from previous page)

```
    %setup
        echo "This is a scriptlet that will be executed on the host, as root, after"
        echo "the container has been bootstrapped. To install things into the container"
        echo "reference the file system location with $SINGULARITY_ROOTFS."

    %files
        /path/on/host/file.txt /path/on/container/file.txt
        relative_file.txt /path/on/container/relative_file.txt

    %post
        echo "This scriptlet section will be executed from within the container after"
        echo "the bootstrap/base has been created and setup."

    %environment
        LUKE=goodguy
        VADER=badguy
        HAN=someguy
        export HAN VADER LUKE

    %test
        echo "Define any test commands that should be executed after container has been"
        echo "built. This scriptlet will be executed from within the running container"
        echo "as the root user. Pay attention to the exit/return value of this scriptlet"
        echo "as any non-zero exit code will be assumed as failure."
        exit 0

    %runscript
        echo "Define actions for the container to be executed with the run command or"
        echo "when container is executed."

    %startscript
        echo "Define actions for container to perform when started as an instance."

    %labels
        HELLO MOTO
        KEY VALUE

    %help
        This is a text file to be displayed with the run-help command.

COMMANDS:

    Build a sif file from a Singularity recipe file:
        $ singularity build /tmp/debian0.sif /path/to/debian.def

    Build a sif image from the Library:
        $ singularity build /tmp/debian1.sif library://debian:latest

    Build a base sandbox from DockerHub, make changes to it, then build sif
        $ singularity build --sandbox /tmp/debian docker://debian:latest
        $ singularity exec --writable /tmp/debian apt-get install python
        $ singularity build /tmp/debian2.sif /tmp/debian
```

Options

```
    --arch string              architecture for remote build (default "amd64")
-B, --bind strings             a user-bind path specification. spec has the format␣
↪src[:dest[:opts]],where src and dest are outside and inside paths. If dest is not␣
↪given,it is set equal to src. Mount options ('opts') may be specified as 'ro'(read-
↪only) or 'rw' (read/write, which is the default).Multiple bind paths can be given by a␣
↪comma separated list. (not supported with remote build)
    --build-arg strings        defines variable=value to replace {{ variable }} entries␣
↪in build definition file
    --build-arg-file string    specifies a file containing variable=value lines to␣
↪replace '{{ variable }}' with value in build definition files
    --builder string           remote Build Service URL, setting this implies --remote
-d, --detached                 submit build job and print build ID (no real-time logs and␣
↪requires --remote)
    --disable-cache            do not use cache or create cache
    --docker-host string       specify a custom Docker daemon host
    --docker-login             login to a Docker Repository interactively
-e, --encrypt                  build an image with an encrypted file system
-f, --fakeroot                 build using user namespace to fake root user (requires a␣
↪privileged installation)
    --fix-perms                ensure owner has rwX permissions on all container content␣
↪for oci/docker sources
-F, --force                    overwrite an image file if it exists
-h, --help                     help for build
    --json                     interpret build definition as JSON
    --library string           container Library URL
    --mount stringArray        a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
    --no-cleanup               do NOT clean up bundle after failed build, can be helpful␣
↪for debugging
    --no-https                 use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-oci                   Launch container with native runtime
    --no-setgroups             disable setgroups when entering --fakeroot user namespace
-T, --notest                   build without running tests in %test section
    --nv                       inject host Nvidia libraries during build for post and␣
↪test sections (not supported with remote build)
    --nvccli                   use nvidia-container-cli for GPU setup (experimental)
    --oci                      Launch container with OCI runtime (experimental)
    --passphrase               prompt for an encryption passphrase
    --pem-path string          enter an path to a PEM formatted RSA key for an encrypted␣
↪container
-r, --remote                   build image remotely (does not require root)
    --rocm                     inject host Rocm libraries during build for post and test␣
↪sections (not supported with remote build)
-s, --sandbox                  build image as sandbox format (chroot directory structure)
    --section strings          only run specific section(s) of deffile (setup, post,␣
↪files, environment, test, labels, none) (default [all])
-u, --update                   run definition over existing container (skips header)
    --writable-tmpfs           during the %test section, makes the file system accessible␣
↪as read-write with non persistent data (with overlay support only)
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.3 singularity cache

Manage the local cache

Synopsis

Manage your local Singularity cache. You can list/clean using the specific types.

```
singularity cache
```

Examples

```
All group commands have their own help output:

$ singularity cache
$ singularity cache --help
```

Options

```
-h, --help   help for cache
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity cache clean* - Clean your local Singularity cache * *singularity cache list* - List your local Singularity cache

Auto generated by spf13/cobra on 23-Nov-2023

8.2.4 singularity cache clean

Clean your local Singularity cache

Synopsis

This will clean your local cache (stored at $HOME/.singularity/cache if SINGULARITY_CACHEDIR is not set). By default the entire cache is cleaned, use –days and –type flags to override this behavior. Note: if you use Singularity as root, cache will be stored in '/root/.singularity/.cache', to clean that cache, you will need to run 'cache clean' as root, or with 'sudo'.

```
singularity cache clean [clean options...]
```

Examples

```
All group commands have their own help output:

$ singularity help cache clean --days 30
$ singularity help cache clean --type=library,oci
$ singularity cache clean --help
```

Options

```
-D, --days int       remove all cache entries older than specified number of days
-n, --dry-run        operate in dry run mode and do not actually clean the cache
-f, --force          suppress any prompts and clean the cache
-h, --help           help for clean
-T, --type strings   a list of cache types to clean, possible entries: all, library, oci-
↪tmp, shub, oras, net, oci-sif, blob (default [all])
```

SEE ALSO

- *singularity cache* - Manage the local cache

Auto generated by spf13/cobra on 23-Nov-2023

8.2.5 singularity cache list

List your local Singularity cache

Synopsis

This will list your local cache (stored at $HOME/.singularity/cache if SINGULARITY_CACHEDIR is not set).

```
singularity cache list [list options...]
```

Examples

```
All group commands have their own help output:

$ singularity help cache list
$ singularity help cache list --type=library,oci
$ singularity cache list --help
```

Options

```
-h, --help           help for list
-T, --type strings   a list of cache types to display, possible entries: all, library,␣
↪oci-tmp, shub, oras, net, oci-sif, blob (default [all])
-v, --verbose        include cache entries in the output
```

SEE ALSO

- *singularity cache* - Manage the local cache

Auto generated by spf13/cobra on 23-Nov-2023

8.2.6 singularity capability

Manage Linux capabilities for users and groups

Synopsis

Capabilities allow you to have fine grained control over the permissions that your containers need to run.

NOTE: capability add/drop commands require root to run. Granting capabilities to users allows them to escalate privilege inside the container and will likely give them a route to privilege escalation on the host system as well. Do not add capabilities to users who should not have root on the host system.

```
singularity capability
```

Examples

```
All group commands have their own help output:

$ singularity help capability add
$ singularity capability add --help
```

Options

```
-h, --help   help for capability
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity capability add* - Add capabilities to a user or group (requires root) * *singularity capability avail* - Show description for available capabilities * *singularity capability drop* - Remove capabilities from a user or group (requires root) * *singularity capability list* - Show capabilities for a given user or group

Auto generated by spf13/cobra on 23-Nov-2023

8.2.7 singularity capability add

Add capabilities to a user or group (requires root)

Synopsis

Add Linux capabilities to a user or group. NOTE: This command requires root to run.

The capabilities argument must be separated by commas and is not case sensitive.

To see available capabilities, type "singularity capability avail" or refer to capabilities manual "man 7 capabilities".

```
singularity capability add [add options...] <capabilities>
```

Examples

```
$ sudo singularity capability add --user nobody AUDIT_READ,chown
$ sudo singularity capability add --group nobody cap_audit_write

To add all capabilities to a user:

$ sudo singularity capability add --user nobody all
```

Options

```
-g, --group string   manage capabilities for a group
-h, --help           help for add
-u, --user string    manage capabilities for a user
```

SEE ALSO

- *singularity capability* - Manage Linux capabilities for users and groups

Auto generated by spf13/cobra on 23-Nov-2023

8.2.8 singularity capability avail

Show description for available capabilities

Synopsis

Show description for available Linux capabilities.

```
singularity capability avail [capabilities]
```

Examples

```
Show description for all available capabilities:

$ singularity capability avail

Show CAP_CHOWN description:

$ singularity capability avail CAP_CHOWN

Show CAP_CHOWN/CAP_NET_RAW description:

$ singularity capability avail CAP_CHOWN,CAP_NET_RAW
```

Options

```
-h, --help   help for avail
```

SEE ALSO

- *singularity capability* - Manage Linux capabilities for users and groups

Auto generated by spf13/cobra on 23-Nov-2023

8.2.9 singularity capability drop

Remove capabilities from a user or group (requires root)

Synopsis

Remove Linux capabilities from a user/group. NOTE: This command requires root to run.

The capabilities argument must be separated by commas and is not case sensitive.

To see available capabilities, type "singularity capability avail" or refer to capabilities manual "man 7 capabilities"

```
singularity capability drop [drop options...] <capabilities>
```

Examples

```
$ sudo singularity capability drop --user nobody AUDIT_READ,CHOWN
$ sudo singularity capability drop --group nobody audit_write

To drop all capabilities for a user:

$ sudo singularity capability drop --user nobody all
```

Options

```
-g, --group string   manage capabilities for a group
-h, --help           help for drop
-u, --user string    manage capabilities for a user
```

SEE ALSO

- *singularity capability* - Manage Linux capabilities for users and groups

Auto generated by spf13/cobra on 23-Nov-2023

8.2.10 singularity capability list

Show capabilities for a given user or group

Synopsis

Show the capabilities for a user or group.

```
singularity capability list [user/group]
```

Examples

```
To list capabilities set for user or group nobody:

$ singularity capability list nobody

To list capabilities for all users/groups:

$ singularity capability list
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity capability* - Manage Linux capabilities for users and groups

Auto generated by spf13/cobra on 23-Nov-2023

8.2.11 singularity config

Manage various singularity configuration (root user only)

Synopsis

The config command allows root user to manage various configuration like fakeroot user mapping entries.

Examples

```
All config commands have their own help output:

$ singularity help config fakeroot
$ singularity config fakeroot --help
```

Options

```
-h, --help   help for config
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity config fakeroot* - Manage fakeroot user mappings entries (root user only) * *singularity config global* - Edit singularity.conf from command line (root user only or unprivileged installation)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.12 singularity config fakeroot

Manage fakeroot user mappings entries (root user only)

Synopsis

The config fakeroot command allow a root user to add/remove/enable/disable fakeroot user mappings.

```
singularity config fakeroot <option> <user>
```

Examples

```
To add a fakeroot user mapping for vagrant user:
$ singularity config fakeroot --add vagrant

To remove a fakeroot user mapping for vagrant user:
$ singularity config fakeroot --remove vagrant

To disable a fakeroot user mapping for vagrant user:
$ singularity config fakeroot --disable vagrant

To enable a fakeroot user mapping for vagrant user:
$ singularity config fakeroot --enable vagrant
```

Options

```
-a, --add       add a fakeroot mapping entry for a user allowing him to use the fakeroot␣
↪feature
-d, --disable   disable a user fakeroot mapping entry preventing him to use the fakeroot␣
↪feature (the user mapping must be present)
-e, --enable    enable a user fakeroot mapping entry allowing him to use the fakeroot␣
↪feature (the user mapping must be present)
-h, --help      help for fakeroot
-r, --remove    remove the user fakeroot mapping entry preventing him to use the␣
↪fakeroot feature
```

SEE ALSO

- *singularity config* - Manage various singularity configuration (root user only)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.13 singularity config global

Edit singularity.conf from command line (root user only or unprivileged installation)

Synopsis

The config global command allow administrators to set/unset/get/reset configuration directives of singularity.conf from command line.

```
singularity config global <option> <directive> [value,...]
```

Examples

```
To add a path to "bind path" directive:
$ singularity config global --set "bind path" /etc/resolv.conf

To remove a path from "bind path" directive:
$ singularity config global --unset "bind path" /etc/resolv.conf

To set "bind path" to the default value:
$ singularity config global --reset "bind path"

To get "bind path" directive value:
$ singularity config global --get "bind path"

To display the resulting configuration instead of writing it to file:
$ singularity config global --dry-run --set "bind path" /etc/resolv.conf
```

Options

```
-d, --dry-run   dump resulting configuration on stdout but doesn't write it to␣
↪singularity.conf
-g, --get       get value of the configuration directive
-h, --help      help for global
-r, --reset     reset the configuration directive value to its default value
-s, --set       set value of the configuration directive (for multi-value directives, it␣
↪will add it)
-u, --unset     unset value of the configuration directive (for multi-value directives,␣
↪it will remove matching values)
```

SEE ALSO

- *singularity config* - Manage various singularity configuration (root user only)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.14 singularity delete

Deletes requested image from the library

Synopsis

The 'delete' command allows you to delete an image from a remote library.

```
singularity delete [arch] <imageRef> [flags]
```

Examples

```
$ singularity delete --arch=amd64 library://username/project/image:1.0
```

Options

```
-A, --arch string      specify requested image arch (default "amd64")
-F, --force            delete image without confirmation
-h, --help             help for delete
    --library string   delete images from the provided library
    --no-https         use http instead of https for docker:// oras:// and library://
↪<hostname>/... URIs
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.15 singularity exec

Run a command within a container

Synopsis

singularity exec supports the following formats:

*.sif Singularity Image Format (SIF). Native to Singularity 3.0+

*.sqsh SquashFS format. Native to Singularity 2.4+

*.img ext3 format. Native to Singularity versions < 2.4.

directory/ sandbox format. Directory containing a valid root file
system and optionally Singularity meta-data.

instance://* A local running instance of a container. (See the instance
command group.)

library://* A SIF container hosted on a Library
(default https://cloud.sylabs.io/library)

docker://* A Docker/OCI container hosted on Docker Hub or another
OCI registry.

shub://* A container hosted on Singularity Hub.

oras://* A SIF container hosted on an OCI registry that supports
the OCI Registry As Storage (ORAS) specification.

```
singularity exec [exec options...] <container> <command>
```

Examples

```
$ singularity exec /tmp/debian.sif cat /etc/debian_version
$ singularity exec /tmp/debian.sif python ./hello_world.py
$ cat hello_world.py | singularity exec /tmp/debian.sif python
$ sudo singularity exec --writable /tmp/debian.sif apt-get update
$ singularity exec instance://my_instance ps -ef
$ singularity exec library://centos cat /etc/os-release
```

Options

```
    --add-caps string               a comma separated capability list to add
    --allow-setuid                  allow setuid binaries in container (root only)
    --app string                    set an application to run inside a container
    --apply-cgroups string          apply cgroups from file for container processes␣
↪(root only)
-B, --bind strings                  a user-bind path specification. spec has the format␣
↪src[:dest[:opts]], where src and dest are outside and inside paths. If dest is not␣
↪given, it is set equal to src. Mount options ('opts') may be specified as 'ro' (read-
↪only) or 'rw' (read/write, which is the default). Multiple bind paths can be given by␣
↪a comma separated list.
    --blkio-weight int              Block IO relative weight in range 10-1000, 0 to␣
↪disable
    --blkio-weight-device strings   Device specific block IO relative weight
    --cdi-dirs strings              comma-separated list of directories in which CDI␣
↪should look for device definition JSON files. If omitted, default will be: /etc/cdi,/
```

(continues on next page)

(continued from previous page)

```
↪var/run/cdi
-e, --cleanenv                      clean environment before running container
    --compat                        apply settings for increased OCI/Docker␣
↪compatibility. Infers --containall, --no-init, --no-umask, --no-eval, --writable-tmpfs.
-c, --contain                       use minimal /dev and empty other directories (e.g. /
↪tmp and $HOME) instead of sharing filesystems from your host
-C, --containall                    contain not only file systems, but also PID, IPC,␣
↪and environment
    --cpu-shares int                CPU shares for container (default -1)
    --cpus string                   Number of CPUs available to container
    --cpuset-cpus string            List of host CPUs available to container
    --cpuset-mems string            List of host memory nodes available to container
    --cwd string                    initial working directory for payload process inside␣
↪the container (synonym for --pwd)
    --device strings                fully-qualified CDI device name(s). A fully-
↪qualified CDI device name consists of a VENDOR, CLASS, and NAME, which are combined as␣
↪follows: <VENDOR>/<CLASS>=<NAME> (e.g. vendor.com/device=mydevice). Multiple fully-
↪qualified CDI device names can be given as a comma separated list.
    --disable-cache                 dont use cache, and dont create cache
    --dns string                    list of DNS server separated by commas to add in␣
↪resolv.conf
    --docker-host string            specify a custom Docker daemon host
    --docker-login                  login to a Docker Repository interactively
    --drop-caps string              a comma separated capability list to drop
    --env stringToString            pass environment variable to contained process␣
↪(default [])
    --env-file string               pass environment variables from file to contained␣
↪process
-f, --fakeroot                      run container in new user namespace as uid 0
    --fusemount strings             A FUSE filesystem mount specification of the form '
↪<type>:<fuse command> <mountpoint>' - where <type> is 'container' or 'host',␣
↪specifying where the mount will be performed ('container-daemon' or 'host-daemon' will␣
↪run the FUSE process detached). <fuse command> is the path to the FUSE executable,␣
↪plus options for the mount. <mountpoint> is the location in the container to which the␣
↪FUSE mount will be attached. E.g. 'container:sshfs 10.0.0.1:/ /sshfs'. Implies --pid.
-h, --help                          help for exec
-H, --home string                   a home directory specification. spec can either be a␣
↪src path or src:dest pair. src is the source path of the home directory outside the␣
↪container and dest overrides the home directory within the container. (default "/home/
↪circleci")
    --hostname string               set container hostname. Infers --uts.
-i, --ipc                           run container in a new IPC namespace
    --keep-privs                    let root user keep privileges in container (root␣
↪only)
    --memory string                 Memory limit in bytes
    --memory-reservation string     Memory soft limit in bytes
    --memory-swap string            Swap limit, use -1 for unlimited swap
    --mount stringArray             a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
-n, --net                           run container in a new network namespace (sets up a␣
↪bridge network interface by default)
    --network string                specify desired network type separated by commas,␣
```

(continues on next page)

(continued from previous page)

```
↪each network will bring up a dedicated interface inside container (default "bridge")
    --network-args strings          specify network arguments to pass to CNI plugins
    --no-compat                     (--oci mode) do not apply settings for increased OCI/
↪Docker compatibility. Emulate native runtime defaults without --contain etc.
    --no-eval                       do not shell evaluate env vars or OCI container CMD/
↪ENTRYPOINT/ARGS
    --no-home                       do NOT mount users home directory if /home is not␣
↪the current working directory
    --no-https                      use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-init                       do NOT start shim process with --pid
    --no-mount strings              disable one or more 'mount xxx' options set in␣
↪singularity.conf, specify absolute destination path to disable a bind path entry, or
↪'bind-paths' to disable all bind path entries.
    --no-oci                        Launch container with native runtime
    --no-pid                        do not run container in a new PID namespace
    --no-privs                      drop all privileges in container (root only in non-
↪OCI mode)
    --no-setgroups                  disable setgroups when entering --fakeroot user␣
↪namespace
    --no-tmp-sandbox                Prohibits unpacking of images into temporary sandbox␣
↪dirs
    --no-umask                      do not propagate umask to the container, set default␣
↪0022 umask
    --nv                            enable Nvidia support
    --nvccli                        use nvidia-container-cli for GPU setup (experimental)
    --oci                           Launch container with OCI runtime (experimental)
    --oom-kill-disable              Disable OOM killer
-o, --overlay strings               use an overlayFS image for persistent data storage␣
↪or as read-only layer of container
    --passphrase                    prompt for an encryption passphrase
    --pem-path string               enter an path to a PEM formatted RSA key for an␣
↪encrypted container
-p, --pid                           run container in a new PID namespace
    --pids-limit int                Limit number of container PIDs, use -1 for unlimited
    --rocm                          enable experimental Rocm support
-S, --scratch strings               include a scratch directory within the container␣
↪that is linked to a temporary dir (use -W to force location)
    --security strings              enable security features (SELinux, Apparmor, Seccomp)
    --sif-fuse                      attempt FUSE mount of SIF (unprivileged / user␣
↪namespace only) (experimental)
-u, --userns                        run container in a new user namespace, allowing␣
↪Singularity to run completely unprivileged on recent kernels. This disables some␣
↪features of Singularity, for example it only works with sandbox images.
    --uts                           run container in a new UTS namespace
-W, --workdir string                working directory to be used for /tmp and /var/tmp␣
↪(if -c/--contain was also used)
-w, --writable                      by default all Singularity containers are available␣
↪as read only. This option makes the file system accessible as read/write.
    --writable-tmpfs                makes the file system accessible as read-write with␣
↪non persistent data (with overlay support only)
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.16 singularity inspect

Show metadata for an image

Synopsis

Inspect will show you labels, environment variables, apps and scripts associated with the image determined by the flags you pass. By default, they will be shown in plain text. If you would like to list them in json format, you should use the –json flag.

```
singularity inspect [inspect options...] <image path>
```

Examples

```
$ singularity inspect ubuntu.sif

If you want to list the applications (apps) installed in a container (located at
/scif/apps) you should run inspect command with --list-apps <container-image> flag.
( See https://sci-f.github.io for more information on SCIF apps)

The following environment variables are available to you when called
from the shell inside the container. The top variables are relevant
to the active app (--app <app>) and the bottom available for all
apps regardless of the active app. Both sets of variables are also available during␣
↪development (at build time).

ACTIVE APP ENVIRONMENT:
    SCIF_APPNAME       the name for the active application
    SCIF_APPROOT       the installation folder for the application created at /scif/apps/
↪<app>
    SCIF_APPMETA       the application metadata folder
    SCIF_APPDATA       the data folder created for the application at /scif/data/<app>
      SCIF_APPINPUT    expected input folder within data base folder
      SCIF_APPOUTPUT   the output data folder within data base folder

    SCIF_APPENV        points to the application's custom environment.sh file in its␣
↪metadata folder
    SCIF_APPLABELS     is the application's labels.json in the metadata folder
    SCIF_APPBIN        is the bin folder for the app, which is automatically added to␣
↪the $PATH when the app is active
    SCIF_APPLIB        is the application's library folder that is added to the LD_
↪LIBRARY_PATH
```

(continues on next page)

(continued from previous page)

```
    SCIF_APPRUN        is the runscript
    SCIF_APPSTART      is the startscript
    SCIF_APPHELP       is the help file for the runscript
    SCIF_APPTEST       is the testing script (test.sh) associated with the application
    SCIF_APPNAME       the name for the active application
    SCIF_APPFILES      the files section associated with the application that are added␣
↪to


GLOBAL APP ENVIRONMENT:

    SCIF_DATA             scif defined data base for all apps (/scif/data)
    SCIF_APPS             scif defined install bases for all apps (/scif/apps)
    SCIF_APPROOT_<app>    root for application <app>
    SCIF_APPDATA_<app>    data root for application <app>

To list all your apps:

$ singularity inspect --list-apps ubuntu.sif

To list only labels in the json format from an image:

$ singularity inspect --json --labels ubuntu.sif

To verify you own a single application on your container image, use the --app <appname>␣
↪flag:

$ singularity inspect --app <appname> ubuntu.sif
```

Options

```
    --all           show all available data (imply --json option)
    --app string    inspect a specific app
-d, --deffile       show the Singularity recipe file that was used to generate the image
-e, --environment   show the environment settings for the image
-h, --help          help for inspect
-H, --helpfile      inspect the runscript helpfile, if it exists
-j, --json          print structured json instead of sections
-l, --labels        show the labels for the image (default)
    --list-apps     list all apps in a container
-r, --runscript     show the runscript for the image
-s, --startscript   show the startscript for the image
-t, --test          show the test script for the image
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.17 singularity instance

Manage containers running as services

Synopsis

Instances allow you to run containers as background processes. This can be useful for running services such as web servers or databases.

```
singularity instance
```

Examples

```
All group commands have their own help output:

$ singularity help instance start
$ singularity instance start --help
```

Options

```
-h, --help   help for instance
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity instance list* - List all running and named Singularity instances * *singularity instance start* - Start a named instance of the given container image * *singularity instance stats* - Get stats for a named instance * *singularity instance stop* - Stop a named instance of a given container image

Auto generated by spf13/cobra on 23-Nov-2023

8.2.18 singularity instance list

List all running and named Singularity instances

Synopsis

The instance list command allows you to view the Singularity container instances that are currently running in the background.

```
singularity instance list [list options...] [<instance name glob>]
```

Examples

```
$ singularity instance list
INSTANCE NAME      PID       IMAGE
test               11963     /home/mibauer/singularity/sinstance/test.sif
test2              11964     /home/mibauer/singularity/sinstance/test.sif
lolcow             11965     /home/mibauer/singularity/sinstance/lolcow.sif

$ singularity instance list 'test*'
INSTANCE NAME      PID       IMAGE
test               11963     /home/mibauer/singularity/sinstance/test.sif
test2              11964     /home/mibauer/singularity/sinstance/test.sif

$ sudo singularity instance list -u mibauer
INSTANCE NAME      PID       IMAGE
test               11963     /home/mibauer/singularity/sinstance/test.sif
test2              16219     /home/mibauer/singularity/sinstance/test.sif
```

Options

```
-h, --help          help for list
-j, --json          print structured json instead of list
-l, --logs          display location of stdout and sterr log files for instances
-u, --user string   if running as root, list instances from "<username>"
```

SEE ALSO

- *singularity instance* - Manage containers running as services

Auto generated by spf13/cobra on 23-Nov-2023

8.2.19 singularity instance start

Start a named instance of the given container image

Synopsis

The instance start command allows you to create a new named instance from an existing container image that will begin running in the background. If a startscript is defined in the container metadata the commands in that script will be executed with the instance start command as well. You can optionally pass arguments to startscript

singularity instance start accepts the following container formats

*.sif Singularity Image Format (SIF). Native to Singularity 3.0+

*.sqsh SquashFS format. Native to Singularity 2.4+

*.img ext3 format. Native to Singularity versions < 2.4.

directory/ sandbox format. Directory containing a valid root file
system and optionally Singularity meta-data.

instance://* A local running instance of a container. (See the instance
command group.)

library://* A SIF container hosted on a Library
(default https://cloud.sylabs.io/library)

docker://* A Docker/OCI container hosted on Docker Hub or another
OCI registry.

shub://* A container hosted on Singularity Hub.

oras://* A SIF container hosted on an OCI registry that supports
the OCI Registry As Storage (ORAS) specification.

```
singularity instance start [start options...] <container path> <instance name>␣
↪[startscript args...]
```

Examples

```
$ singularity instance start /tmp/my-sql.sif mysql

$ singularity shell instance://mysql
Singularity my-sql.sif> pwd
/home/mibauer/mysql
Singularity my-sql.sif> ps
PID TTY          TIME CMD
  1 pts/0    00:00:00 sinit
  2 pts/0    00:00:00 bash
  3 pts/0    00:00:00 ps
Singularity my-sql.sif>

$ singularity instance stop /tmp/my-sql.sif mysql
Stopping /tmp/my-sql.sif mysql
```

Options

```
    --add-caps string               a comma separated capability list to add
    --allow-setuid                  allow setuid binaries in container (root only)
    --app string                    set an application to run inside a container
    --apply-cgroups string          apply cgroups from file for container processes␣
↪(root only)
-B, --bind strings                  a user-bind path specification. spec has the format␣
↪src[:dest[:opts]], where src and dest are outside and inside paths. If dest is not␣
↪given, it is set equal to src. Mount options ('opts') may be specified as 'ro' (read-
↪only) or 'rw' (read/write, which is the default). Multiple bind paths can be given by␣
↪a comma separated list.
    --blkio-weight int              Block IO relative weight in range 10-1000, 0 to␣
↪disable
    --blkio-weight-device strings   Device specific block IO relative weight
    --boot                          execute /sbin/init to boot container (root only)
-e, --cleanenv                      clean environment before running container
    --compat                        apply settings for increased OCI/Docker␣
↪compatibility. Infers --containall, --no-init, --no-umask, --no-eval, --writable-tmpfs.
-c, --contain                       use minimal /dev and empty other directories (e.g. /
↪tmp and $HOME) instead of sharing filesystems from your host
-C, --containall                    contain not only file systems, but also PID, IPC,␣
↪and environment
    --cpu-shares int                CPU shares for container (default -1)
    --cpus string                   Number of CPUs available to container
    --cpuset-cpus string            List of host CPUs available to container
    --cpuset-mems string            List of host memory nodes available to container
    --disable-cache                 dont use cache, and dont create cache
    --dns string                    list of DNS server separated by commas to add in␣
↪resolv.conf
    --docker-host string            specify a custom Docker daemon host
    --docker-login                  login to a Docker Repository interactively
    --drop-caps string              a comma separated capability list to drop
    --env stringToString            pass environment variable to contained process␣
↪(default [])
    --env-file string               pass environment variables from file to contained␣
↪process
-f, --fakeroot                      run container in new user namespace as uid 0
    --fusemount strings             A FUSE filesystem mount specification of the form '
↪<type>:<fuse command> <mountpoint>' - where <type> is 'container' or 'host',␣
↪specifying where the mount will be performed ('container-daemon' or 'host-daemon' will␣
↪run the FUSE process detached). <fuse command> is the path to the FUSE executable,␣
↪plus options for the mount. <mountpoint> is the location in the container to which the␣
↪FUSE mount will be attached. E.g. 'container:sshfs 10.0.0.1:/ /sshfs'. Implies --pid.
-h, --help                          help for start
-H, --home string                   a home directory specification. spec can either be a␣
↪src path or src:dest pair. src is the source path of the home directory outside the␣
↪container and dest overrides the home directory within the container. (default "/home/
↪circleci")
    --hostname string               set container hostname. Infers --uts.
-i, --ipc                           run container in a new IPC namespace
    --keep-privs                    let root user keep privileges in container (root␣
↪only)
```

(continues on next page)

(continued from previous page)

```
    --memory string                 Memory limit in bytes
    --memory-reservation string     Memory soft limit in bytes
    --memory-swap string            Swap limit, use -1 for unlimited swap
    --mount stringArray             a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
-n, --net                           run container in a new network namespace (sets up a␣
↪bridge network interface by default)
    --network string                specify desired network type separated by commas,␣
↪each network will bring up a dedicated interface inside container (default "bridge")
    --network-args strings          specify network arguments to pass to CNI plugins
    --no-compat                     (--oci mode) do not apply settings for increased OCI/
↪Docker compatibility. Emulate native runtime defaults without --contain etc.
    --no-eval                       do not shell evaluate env vars or OCI container CMD/
↪ENTRYPOINT/ARGS
    --no-home                       do NOT mount users home directory if /home is not␣
↪the current working directory
    --no-https                      use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-init                       do NOT start shim process with --pid
    --no-mount strings              disable one or more 'mount xxx' options set in␣
↪singularity.conf, specify absolute destination path to disable a bind path entry, or
↪'bind-paths' to disable all bind path entries.
    --no-oci                        Launch container with native runtime
    --no-privs                      drop all privileges in container (root only in non-
↪OCI mode)
    --no-setgroups                  disable setgroups when entering --fakeroot user␣
↪namespace
    --no-tmp-sandbox                Prohibits unpacking of images into temporary sandbox␣
↪dirs
    --no-umask                      do not propagate umask to the container, set default␣
↪0022 umask
    --nv                            enable Nvidia support
    --nvccli                        use nvidia-container-cli for GPU setup (experimental)
    --oci                           Launch container with OCI runtime (experimental)
    --oom-kill-disable              Disable OOM killer
-o, --overlay strings               use an overlayFS image for persistent data storage␣
↪or as read-only layer of container
    --passphrase                    prompt for an encryption passphrase
    --pem-path string               enter an path to a PEM formatted RSA key for an␣
↪encrypted container
    --pid-file string               write instance PID to the file with the given name
    --pids-limit int                Limit number of container PIDs, use -1 for unlimited
    --rocm                          enable experimental Rocm support
-S, --scratch strings               include a scratch directory within the container␣
↪that is linked to a temporary dir (use -W to force location)
    --security strings              enable security features (SELinux, Apparmor, Seccomp)
-u, --userns                        run container in a new user namespace, allowing␣
↪Singularity to run completely unprivileged on recent kernels. This disables some␣
↪features of Singularity, for example it only works with sandbox images.
    --uts                           run container in a new UTS namespace
-W, --workdir string                working directory to be used for /tmp and /var/tmp␣
↪(if -c/--contain was also used)
```

(continues on next page)

(continued from previous page)

```
-w, --writable                      by default all Singularity containers are available␣
↪as read only. This option makes the file system accessible as read/write.
    --writable-tmpfs                makes the file system accessible as read-write with␣
↪non persistent data (with overlay support only)
```

SEE ALSO

- *singularity instance* - Manage containers running as services

Auto generated by spf13/cobra on 23-Nov-2023

8.2.20 singularity instance stats

Get stats for a named instance

Synopsis

The instance stats command allows you to get statistics for a named instance, either printed to the terminal or in json. If you are root, you can optionally ask for statistics for a container instance belonging to a specific user. If you add –no-stream, you will only see one timepoint. Asking for json implies the same.

```
singularity instance stats [stats options...] <instance name>
```

Examples

```
$ singularity instance stats mysql
$ singularity instance stats --json mysql
$ singularity instance stats --no-stream mysql
$ sudo singularity instance stats --user <username> user-mysql
```

Options

```
-h, --help          help for stats
-j, --json          output stats in json
    --no-stream     disable streaming (live update) of instance stats
-u, --user string   view stats for an instance belonging to a user (root only)
```

SEE ALSO

- *singularity instance* - Manage containers running as services

Auto generated by spf13/cobra on 23-Nov-2023

8.2.21 singularity instance stop

Stop a named instance of a given container image

Synopsis

The command singularity instance stop allows you to stop and clean up a named, running instance of a given container image.

```
singularity instance stop [stop options...] [instance]
```

Examples

```
$ singularity instance start my-sql.sif mysql1
$ singularity instance start my-sql.sif mysql2
$ singularity instance stop mysql*
Stopping mysql1 instance of my-sql.sif (PID=23845)
Stopping mysql2 instance of my-sql.sif (PID=23858)

$ singularity instance start my-sql.sif mysql1

Force instance to shutdown
$ singularity instance stop -f mysql1 (may corrupt data)

Send SIGTERM to the instance
$ singularity instance stop -s SIGTERM mysql1
$ singularity instance stop -s TERM mysql1
$ singularity instance stop -s 15 mysql1
```

Options

```
-a, --all             stop all user's instances
-F, --force           force kill instance
-h, --help            help for stop
-s, --signal string   signal sent to the instance
-t, --timeout int     force kill non stopped instances after X seconds (default 10)
-u, --user string     if running as root, stop instances belonging to user
```

SEE ALSO

- *singularity instance* - Manage containers running as services

Auto generated by spf13/cobra on 23-Nov-2023

8.2.22 singularity key

Manage OpenPGP keys

Synopsis

Manage your trusted, public and private keys in your local or in the global keyring (local keyring: '~/.singularity/sypgp' if 'SINGULARITY_SYPGPDIR' is not set, global keyring: '/usr/local/etc/singularity/global-pgp-public')

```
singularity key [key options...]
```

Examples

```
All group commands have their own help output:

$ singularity help key newpair
$ singularity key list --help
```

Options

```
-h, --help   help for key
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity key export* - Export a public or private key into a specific file * *singularity key import* - Import a local key into the local or global keyring * *singularity key list* - List keys in your local or in the global keyring * *singularity key newpair* - Create a new key pair * *singularity key pull* - Download a public key from a key server * *singularity key push* - Upload a public key to a key server * *singularity key remove* - Remove a local public key from your local or the global keyring * *singularity key search* - Search for keys on a key server

Auto generated by spf13/cobra on 23-Nov-2023

8.2.23 singularity key export

Export a public or private key into a specific file

Synopsis

The 'key export' command allows you to export a key and save it to a file.

```
singularity key export [export options...] <output-file>
```

Examples

```
Exporting a private key:

$ singularity key export --secret ./private.asc

Exporting a public key:

$ singularity key export ./public.asc
```

Options

```
-a, --armor     ascii armored format
-g, --global    manage global public keys (import/pull/remove are restricted to root␣
↪user or unprivileged installation only)
-h, --help      help for export
    --private   export a private key (synonym for --secret)
-s, --secret    export a secret key (synonym for --private)
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.24 singularity key import

Import a local key into the local or global keyring

Synopsis

The 'key import' command allows you to add a key to your local or global keyring from a specific file.

```
singularity key import [import options...] <input-key>
```

Examples

```
$ singularity key import ./my-key.asc

# Import into global keyring (root user only)
$ singularity key import --global ./my-key.asc
```

Options

```
-g, --global         manage global public keys (import/pull/remove are restricted to␣
↪root user or unprivileged installation only)
-h, --help           help for import
    --new-password   set a new password to the private key
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.25 singularity key list

List keys in your local or in the global keyring

Synopsis

List your local keys in your keyring. Will list public (trusted) keys by default.

```
singularity key list
```

Examples

```
$ singularity key list
$ singularity key list --secret

# list global public keys
$ singularity key list --global
```

Options

```
-g, --global    manage global public keys (import/pull/remove are restricted to root␣
↪user or unprivileged installation only)
-h, --help      help for list
    --private   list private keys instead of the default which displays public ones␣
↪(synonym for --secret)
-s, --secret    list secret keys instead of the default which displays public ones␣
↪(synonym for --private)
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.26 singularity key newpair

Create a new key pair

Synopsis

The 'key newpair' command allows you to create a new key or public/private keys to be stored in the default user local keyring location (e.g., $HOME/.singularity/sypgp).

```
singularity key newpair
```

Examples

```
$ singularity key newpair
$ singularity key newpair --password=psk --name=your-name --comment="key comment" --
↪email=mail@email.com --push=false
```

Options

```
-b, --bit-length int    specify key bit length (default 4096)
-C, --comment string    key comment
-E, --email string      key owner email
-h, --help              help for newpair
-N, --name string       key owner name
-P, --password string   key password
-U, --push              specify to push the public key to the remote keystore (default␣
↪true)
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.27 singularity key pull

Download a public key from a key server

Synopsis

The 'key pull' command allows you to retrieve public key material from a remote key server, and add it to your keyring. Note that Singularity consults your keyring when running commands such as 'singularity verify', and thus adding a key to your keyring implies a level of trust. Because of this, it is recommended that you verify the fingerprint of the key with its owner prior to running this command.

```
singularity key pull [pull options...] <fingerprint>
```

Examples

```
$ singularity key pull 8883491F4268F173C6E5DC49EDECE4F3F38D871E
```

Options

```
-g, --global       manage global public keys (import/pull/remove are restricted to root␣
↪user or unprivileged installation only)
-h, --help         help for pull
-u, --url string   specify the key server URL
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.28 singularity key push

Upload a public key to a key server

Synopsis

The 'key push' command allows you to connect to a key server and upload public keys from the local or the global keyring.

```
singularity key push [push options...] <fingerprint>
```

Examples

```
$ singularity key push 8883491F4268F173C6E5DC49EDECE4F3F38D871E
```

Options

```
-g, --global       manage global public keys (import/pull/remove are restricted to root␣
↪user or unprivileged installation only)
-h, --help         help for push
-u, --url string   specify the key server URL
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.29 singularity key remove

Remove a local public key from your local or the global keyring

Synopsis

The 'key remove' command will remove a local public key from the local or the global keyring.

```
singularity key remove <fingerprint>
```

Examples

```
$ singularity key remove D87FE3AF5C1F063FCBCC9B02F812842B5EEE5934
```

Options

```
-g, --global    manage global public keys (import/pull/remove are restricted to root␣
↪user or unprivileged installation only)
-h, --help      help for remove
    --private   remove a private key (synonym for --secret)
-s, --secret    remove a secret key (synonym for --private)
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.30 singularity key search

Search for keys on a key server

Synopsis

The 'key search' command allows you to connect to a key server and look for public keys matching the argument passed to the command line. You can search by name, email, or fingerprint / key ID. (Maximum 100 search entities)

```
singularity key search [search options...] <search_string>
```

Examples

```
$ singularity key search sylabs.io

# search by fingerprint:
$ singularity key search 8883491F4268F173C6E5DC49EDECE4F3F38D871E

# search by key ID:
$ singularity key search F38D871E
```

Options

```
-h, --help         help for search
-l, --long-list    output long list when searching for keys
-u, --url string   specify the key server URL
```

SEE ALSO

- *singularity key* - Manage OpenPGP keys

Auto generated by spf13/cobra on 23-Nov-2023

8.2.31 singularity keyserver

Manage singularity keyservers

Synopsis

The 'keyserver' command allows you to manage standalone keyservers that will be used for retrieving cryptographic keys.

Examples

```
All group commands have their own help output:

  $ singularity help keyserver add
  $ singularity keyserver add
```

Options

```
-c, --config string   path to the file holding keyserver configurations (default "/home/
↪circleci/.singularity/remote.yaml")
-h, --help            help for keyserver
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity keyserver add* - Add a keyserver (root user only) * *singularity keyserver list* - List all keyservers that are configured * *singularity keyserver login* - Login to a keyserver * *singularity keyserver logout* - Logout from a keyserver * *singularity keyserver remove* - Remove a keyserver (root user only)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.32 singularity keyserver add

Add a keyserver (root user only)

Synopsis

The 'keyserver add' command lets the user specify an additional keyserver. The –order specifies the order of the new keyserver relative to the keyservers that have already been specified. Therefore, when specifying '–order 1', the new keyserver will become the primary one. If no endpoint is specified, the new keyserver will be associated with the default remote endpoint (SylabsCloud).

```
singularity keyserver add [options] [remoteName] <keyserver_url>
```

Examples

```
$ singularity keyserver add https://keys.example.com

To add a keyserver to be used as the primary keyserver for the current
endpoint:
$ singularity keyserver add --order 1 https://keys.example.com
```

Options

```
-h, --help           help for add
-i, --insecure       allow insecure connection to keyserver
-o, --order uint32   define the keyserver order
```

SEE ALSO

- *singularity keyserver* - Manage singularity keyservers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.33 singularity keyserver list

List all keyservers that are configured

Synopsis

The 'keyserver list' command lists the keyservers configured for use with each remote endpoint. If the optional remoteName argument is provided, the command will only list keyservers for the remote endpoint matching that name.

```
singularity keyserver list [remoteName]
```

Examples

```
$ singularity keyserver list
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity keyserver* - Manage singularity keyservers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.34 singularity keyserver login

Login to a keyserver

Synopsis

The 'keyserver login' command allows you to login to a specific keyserver.

```
singularity keyserver login [login options...] <keyserver>
```

Examples

```
To login in to a keyserver:
$ singularity keyserver login --username foo https://mykeyserver.example.com
```

Options

```
-h, --help              help for login
-p, --password string   password / token to authenticate with
    --password-stdin    take password from standard input
-u, --username string   username to authenticate with (required for Docker/OCI registry␣
↪login)
```

SEE ALSO

- *singularity keyserver* - Manage singularity keyservers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.35 singularity keyserver logout

Logout from a keyserver

Synopsis

The 'keyserver logout' command allows you to log out from a keyserver.

```
singularity keyserver logout <keyserver>
```

Examples

```
To log out from a keyserver:
$ singularity keyserver logout https://mykeyserver.example.com
```

Options

```
-h, --help   help for logout
```

SEE ALSO

- *singularity keyserver* - Manage singularity keyservers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.36 singularity keyserver remove

Remove a keyserver (root user only)

Synopsis

The 'keyserver remove' command lets the user remove a previously specified keyserver from a specific endpoint. If no endpoint is specified, the default remote endpoint (SylabsCloud) will be assumed.

```
singularity keyserver remove [remoteName] <keyserver_url>
```

Examples

```
$ singularity keyserver remove https://keys.example.com
```

Options

```
-h, --help   help for remove
```

SEE ALSO

- *singularity keyserver* - Manage singularity keyservers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.37 singularity oci

Manage OCI containers

Synopsis

Allow you to manage containers from OCI bundle directories.

NOTE: all oci commands requires to run as root

Examples

```
All group commands have their own help output:

$ singularity oci create -b ~/bundle mycontainer
$ singularity oci start mycontainer
```

Options

```
-h, --help   help for oci
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity oci attach* - Attach console to a running container process (root user only) * *singularity oci create* - Create a container from a bundle directory (root user only) * *singularity oci delete* - Delete container (root user only) * *singularity oci exec* - Execute a command within container (root user only) * *singularity oci kill* - Kill a container (root user only) * *singularity oci mount* - Mount create an OCI bundle from SIF image (root user only) * *singularity oci pause* - Suspends all processes inside the container (root user only) * *singularity oci resume* - Resumes all processes previously paused inside the container (root user only) * *singularity oci run* - Create/start/attach/delete a container from a bundle directory (root user only) * *singularity oci start* - Start container process (root user only) * *singularity oci state* - Query state of a container (root user only) * *singularity oci umount* - Umount delete bundle (root user only) * *singularity oci update* - Update container cgroups resources (root user only)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.38 singularity oci attach

Attach console to a running container process (root user only)

Synopsis

Attach will attach console to a running container process running within container identified by container ID.

```
singularity oci attach <container_ID>
```

Examples

```
$ singularity oci attach mycontainer
```

Options

```
-h, --help   help for attach
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.39 singularity oci create

Create a container from a bundle directory (root user only)

Synopsis

Create invoke create operation to create a container instance from an OCI bundle directory

```
singularity oci create -b <bundle_path> [create options...] <container_ID>
```

Examples

```
$ singularity oci create -b ~/bundle mycontainer
```

Options

```
-b, --bundle string       specify the OCI bundle path (required)
-h, --help                help for create
    --log-format string   specify the log file format. Available formats are basic,␣
↪kubernetes and json (default "kubernetes")
-l, --log-path string     specify the log file path
    --pid-file string     specify the pid file
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.40 singularity oci delete

Delete container (root user only)

Synopsis

Delete invoke delete operation to delete resources that were created for container identified by container ID.

```
singularity oci delete <container_ID>
```

Examples

```
$ singularity oci delete mycontainer
```

Options

```
-h, --help   help for delete
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.41 singularity oci exec

Execute a command within container (root user only)

Synopsis

Exec will execute the provided command/arguments within container identified by container ID.

```
singularity oci exec <container_ID> <command> <args>
```

Examples

```
$ singularity oci exec mycontainer id
```

Options

```
-h, --help   help for exec
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.42 singularity oci kill

Kill a container (root user only)

Synopsis

Kill invoke kill operation to kill processes running within container identified by container ID.

```
singularity oci kill [kill options...] <container_ID>
```

Examples

```
$ singularity oci kill mycontainer INT
$ singularity oci kill mycontainer -s INT
```

Options

```
-f, --force           kill container process with SIGKILL
-h, --help            help for kill
-s, --signal string   signal sent to the container (default "SIGTERM")
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.43 singularity oci mount

Mount create an OCI bundle from SIF image (root user only)

Synopsis

Mount will mount and create an OCI bundle from a SIF image.

```
singularity oci mount <sif_image> <bundle_path>
```

Examples

```
$ singularity oci mount /tmp/example.sif /var/lib/singularity/bundles/example
```

Options

```
-h, --help   help for mount
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.44 singularity oci pause

Suspends all processes inside the container (root user only)

Synopsis

Pause will suspend all processes for the specified container ID.

```
singularity oci pause <container_ID>
```

Examples

```
$ singularity oci pause mycontainer
```

Options

```
-h, --help   help for pause
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.45 singularity oci resume

Resumes all processes previously paused inside the container (root user only)

Synopsis

Resume will resume all processes previously paused for the specified container ID.

```
singularity oci resume <container_ID>
```

Examples

```
$ singularity oci resume mycontainer
```

Options

```
-h, --help   help for resume
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.46 singularity oci run

Create/start/attach/delete a container from a bundle directory (root user only)

Synopsis

Run will invoke equivalent of create/start/attach/delete commands in a row.

```
singularity oci run -b <bundle_path> [run options...] <container_ID>
```

Examples

```
$ singularity oci run -b ~/bundle mycontainer

is equivalent to :

$ singularity oci create -b ~/bundle mycontainer
$ singularity oci start mycontainer
$ singularity oci attach mycontainer
$ singularity oci delete mycontainer
```

Options

```
-b, --bundle string       specify the OCI bundle path (required)
-h, --help                help for run
    --log-format string   specify the log file format. Available formats are basic,␣
↪kubernetes and json (default "kubernetes")
-l, --log-path string     specify the log file path
    --pid-file string     specify the pid file
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.47 singularity oci start

Start container process (root user only)

Synopsis

Start invoke start operation to start a previously created container identified by container ID.

```
singularity oci start <container_ID>
```

Examples

```
$ singularity oci start mycontainer
```

Options

```
-h, --help   help for start
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.48 singularity oci state

Query state of a container (root user only)

Synopsis

State invoke state operation to query state of a created/running/stopped container identified by container ID.

```
singularity oci state <container_ID>
```

Examples

```
$ singularity oci state mycontainer
```

Options

```
-h, --help   help for state
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.49 singularity oci umount

Umount delete bundle (root user only)

Synopsis

Umount will umount an OCI bundle previously mounted with singularity oci mount.

```
singularity oci umount <bundle_path>
```

Examples

```
$ singularity oci umount /var/lib/singularity/bundles/example
```

Options

```
-h, --help   help for umount
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.50 singularity oci update

Update container cgroups resources (root user only)

Synopsis

Update will update cgroups resources for the specified container ID. Container must be in a RUNNING or CREATED state.

```
singularity oci update [update options...] <container_ID>
```

Examples

```
$ singularity oci update --from-file /tmp/cgroups-update.json mycontainer

or to update from stdin :

$ cat /tmp/cgroups-update.json | singularity oci update --from-file - mycontainer
```

Options

```
-f, --from-file string   specify path to OCI JSON cgroups resource file ('-' to read␣
↪from STDIN)
-h, --help               help for update
```

SEE ALSO

- *singularity oci* - Manage OCI containers

Auto generated by spf13/cobra on 23-Nov-2023

8.2.51 singularity overlay

Manage an EXT3 writable overlay image

Synopsis

The overlay command allows management of EXT3 writable overlay images.

```
singularity overlay
```

Examples

```
All overlay commands have their own help output:

$ singularity help overlay create
$ singularity overlay create --help
```

Options

```
-h, --help   help for overlay
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity overlay create* - Create EXT3 writable overlay image

Auto generated by spf13/cobra on 23-Nov-2023

8.2.52 singularity overlay create

Create EXT3 writable overlay image

Synopsis

The overlay create command allows to create EXT3 writable overlay image either as a single EXT3 image or by adding it automatically to an existing SIF image.

```
singularity overlay create <options> image
```

Examples

```
To create and add a writable overlay to an existing SIF image:
$ singularity overlay create --size 1024 /tmp/image.sif

To create a single EXT3 writable overlay image:
$ singularity overlay create --size 1024 /tmp/my_overlay.img

To create a sparse overlay when creating a new ext3 file system image:
$ singularity overlay create --size 1024 --sparse /tmp/ext3_overlay.img
```

Options

```
    --create-dir strings   directory to create as part of the overlay layout
-h, --help                 help for create
-s, --size int             size of the EXT3 writable overlay in MiB (default 64)
-S, --sparse               create a sparse overlay
```

SEE ALSO

- *singularity overlay* - Manage an EXT3 writable overlay image

Auto generated by spf13/cobra on 23-Nov-2023

8.2.53 singularity plugin

Manage Singularity plugins

Synopsis

The 'plugin' command allows you to manage Singularity plugins which provide add-on functionality to the default Singularity installation.

```
singularity plugin [plugin options...]
```

Examples

```
All group commands have their own help output:

$ singularity help plugin compile
$ singularity plugin list --help
```

Options

```
-h, --help   help for plugin
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity plugin compile* - Compile a Singularity plugin * *singularity plugin create* - Create a plugin skeleton directory * *singularity plugin disable* - disable an installed Singularity plugin * *singularity plugin enable* - Enable an installed Singularity plugin * *singularity plugin inspect* - Inspect a singularity plugin (either an installed one or an image) * *singularity plugin install* - Install a compiled Singularity plugin * *singularity plugin list* - List installed Singularity plugins * *singularity plugin uninstall* - Uninstall removes the named plugin from the system

Auto generated by spf13/cobra on 23-Nov-2023

8.2.54 singularity plugin compile

Compile a Singularity plugin

Synopsis

The 'plugin compile' command allows a developer to compile a Singularity plugin in the expected environment. The provided host directory is the location of the plugin's source code. A compiled plugin is packed into a SIF file.

NOTE: Before using this command, make sure that you trust the origin of the plugin, and that you are certain it does not contain any malicious code.

```
singularity plugin compile [compile options...] <host_path>
```

Examples

```
$ singularity plugin compile $HOME/singularity/test-plugin
```

Options

```
-h, --help         help for compile
-o, --out string   path of the SIF output file
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.55 singularity plugin create

Create a plugin skeleton directory

Synopsis

The 'plugin create' command allows a user to creates a plugin skeleton directory structure to start development of a new plugin.

```
singularity plugin create <host_path> <name>
```

Examples

```
$ singularity plugin create ~/myplugin github.com/username/myplugin
$ ls -1 ~/myplugin
go.mod
main.go
singularity_source
```

Options

```
-h, --help   help for create
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.56 singularity plugin disable

disable an installed Singularity plugin

Synopsis

The 'plugin disable' command allows a user to disable a plugin that is already installed in the system and which has been previously enabled.

```
singularity plugin disable <name>
```

Examples

```
$ singularity plugin disable example.org/plugin
```

Options

```
-h, --help   help for disable
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.57 singularity plugin enable

Enable an installed Singularity plugin

Synopsis

The 'plugin enable' command allows a user to enable a plugin that is already installed in the system and which has been previously disabled.

```
singularity plugin enable <name>
```

Examples

```
$ singularity plugin enable example.org/plugin
```

Options

```
-h, --help   help for enable
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.58 singularity plugin inspect

Inspect a singularity plugin (either an installed one or an image)

Synopsis

The 'plugin inspect' command allows a user to inspect a plugin that is already installed in the system or an image containing a plugin that is yet to be installed.

```
singularity plugin inspect (<name>|<image>)
```

Examples

```
$ singularity plugin inspect sylabs.io/test-plugin
Name: sylabs.io/test-plugin
Description: A test Singularity plugin.
Author: Sylabs
Version: 0.1.0
```

Options

```
-h, --help   help for inspect
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.59 singularity plugin install

Install a compiled Singularity plugin

Synopsis

The 'plugin install' command installs the compiled plugin found at plugin_path into the appropriate directory on the host.

NOTE: Before using this command, make sure that you trust the origin of the plugin, and that you are certain it does not contain any malicious code.

```
singularity plugin install <plugin_path>
```

Examples

```
$ singularity plugin install $HOME/singularity/test-plugin/test-plugin.sif
```

Options

```
-h, --help   help for install
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.60 singularity plugin list

List installed Singularity plugins

Synopsis

The 'plugin list' command lists the Singularity plugins installed on the host.

```
singularity plugin list [list options...]
```

Examples

```
$ singularity plugin list
ENABLED  NAME
    yes  example.org/plugin
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.61 singularity plugin uninstall

Uninstall removes the named plugin from the system

Synopsis

The 'plugin uninstall' command removes the named plugin from the system

```
singularity plugin uninstall <name>
```

Examples

```
$ singularity plugin uninstall example.org/plugin
```

Options

```
-h, --help   help for uninstall
```

SEE ALSO

- *singularity plugin* - Manage Singularity plugins

Auto generated by spf13/cobra on 23-Nov-2023

8.2.62 singularity pull

Pull an image from a URI

Synopsis

The 'pull' command allows you to download or build a container from a given URI. Supported URIs include:

library: Pull an image from the currently configured library
library://user/collection/container[:tag]

docker: Pull a Docker/OCI image from Docker Hub, or another OCI registry.
docker://user/image:tag

shub: Pull an image from Singularity Hub
shub://user/image:tag

oras: Pull an image from an OCI registry that supports ORAS / OCI artifacts.
oras://registry/namespace/image:tag

http, https: Pull an image using the http(s?) protocol
https://example.com/containers/mycontainer.sif

By default, images from a library URI will be pulled in the same format they were uploaded. If the –oci flag is specified then the pull is required to result in an OCI-SIF image.

By default, images pulled from docker and other oci URIs will be converted into a singularity native SIF image. If the –oci flag is specified then they will be encapsulated in an OCI-SIF image.

Images pulled from a shub/oras/http/https URI are always directly downloaded, in the same format as they were uploaded.

```
singularity pull [pull options...] [output file] <URI>
```

Examples

```
From Sylabs cloud library
$ singularity pull alpine.sif library://alpine:latest

From Docker to a singularity native SIF image
$ singularity pull tensorflow.sif docker://tensorflow/tensorflow:latest

From Docker to an OCI-SIF image
$ singularity pull --oci tensorflow.oci.sif docker://tensorflow/tensorflow:latest

From Shub
$ singularity pull singularity-images.sif shub://vsoch/singularity-images

From an OCI registry supporting ORAS / OCI artifacts
$ singularity pull image.sif oras://<username>.azurecr.io/namespace/image:tag
```

Options

```
    --arch string          architecture to use when pulling images
    --dir string           download images to the specific directory
    --disable-cache        dont use cached images/blobs and dont create them
    --docker-host string   specify a custom Docker daemon host
    --docker-login         login to a Docker Repository interactively
-F, --force                overwrite an image file if it exists
-h, --help                 help for pull
    --library string       download images from the provided library
    --no-cleanup           do NOT clean up bundle after failed build, can be helpful for␣
↪debugging
    --no-https             use http instead of https for docker:// oras:// and library://
↪<hostname>/... URIs
    --no-oci               Launch container with native runtime
    --oci                  Launch container with OCI runtime (experimental)
    --platform string      platform (OS/Architecture/Variant) to use when pulling images
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.63 singularity push

Upload image to the provided URI

Synopsis

The 'push' command allows you to upload a SIF container to a given URI. Supported URIs include:

library:
library://user/collection/container[:tag]

oras:
oras://registry/namespace/repo:tag

NOTE: It's always good practice to sign your containers before pushing them to the library. An auth token is required to push to the library, so you may need to configure it first with 'singularity remote'.

```
singularity push [push options...] <image> <URI>
```

Examples

```
To Library
$ singularity push /home/user/my.sif library://user/collection/my.sif:latest

To supported OCI registry
$ singularity push /home/user/my.sif oras://registry/namespace/image:tag
```

Options

```
-U, --allow-unsigned       do not require a signed container image
-D, --description string   description for container image (library:// only)
    --docker-host string   specify a custom Docker daemon host
-h, --help                 help for push
    --library string       the library to push to
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.64 singularity registry

Manage authentication to OCI/Docker registries

Synopsis

The 'registry' command allows you to manage authentication to standalone OCI/Docker registries, such as 'docker://" or 'oras://'.

Examples

```
All group commands have their own help output:

  $ singularity help registry login
  $ singularity registry login
```

Options

```
-c, --config string   path to the file holding registry configurations (default "/home/
↪circleci/.singularity/remote.yaml")
-h, --help            help for registry
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity registry list* - List all OCI credentials that are configured * *singularity registry login* - Login to an OCI/Docker registry * *singularity registry logout* - Logout from an OCI/Docker registry

Auto generated by spf13/cobra on 23-Nov-2023

8.2.65 singularity registry list

List all OCI credentials that are configured

Synopsis

The 'registry list' command lists all credentials for OCI/Docker registries that are configured for use.

```
singularity registry list
```

Examples

```
$ singularity registry list
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity registry* - Manage authentication to OCI/Docker registries

Auto generated by spf13/cobra on 23-Nov-2023

8.2.66 singularity registry login

Login to an OCI/Docker registry

Synopsis

The 'registry login' command allows you to login to a specific OCI/Docker registry.

```
singularity registry login [login options...] <registry_uri>
```

Examples

```
To login in to a docker/OCI registry:
$ singularity registry login --username foo docker://docker.io
$ singularity registry login --username foo oras://myregistry.example.com

Note that many cloud OCI registries use token-based authentication. The token
should be specified as the password for login. A username is still required.
E.g. when using a standard Azure identity and token to login to an ACR
registry, the username '00000000-0000-0000-0000-000000000000' is required.
Consult your provider's documentation for details concerning their specific
login requirements.
```

Options

```
-h, --help              help for login
-p, --password string   password / token to authenticate with
    --password-stdin    take password from standard input
-u, --username string   username to authenticate with (required for Docker/OCI registry␣
↪login)
```

SEE ALSO

- *singularity registry* - Manage authentication to OCI/Docker registries

Auto generated by spf13/cobra on 23-Nov-2023

8.2.67 singularity registry logout

Logout from an OCI/Docker registry

Synopsis

The 'registry logout' command allows you to log out from an OCI/Docker registry.

```
singularity registry logout <registry_uri>
```

Examples

```
To log out from an OCI/Docker registry
$ singularity registry logout docker://docker.io
```

Options

```
-h, --help   help for logout
```

SEE ALSO

- *singularity registry* - Manage authentication to OCI/Docker registries

Auto generated by spf13/cobra on 23-Nov-2023

8.2.68 singularity remote

Manage singularity remote endpoints

Synopsis

The 'remote' command allows you to manage Singularity remote endpoints through its subcommands.

A 'remote endpoint' is the Sylabs Cloud, a Singularity Enterprise installation, or a compatible group of services. The remote endpoint is a single address, e.g. 'cloud.sylabs.io' through which linked library, builder and keystore sevices will be automatically discovered.

To configure a remote endpoint you must 'remote add' it. You can 'remote login' if you will be performing actions needing authentication. Switch between configured remote endpoints with the 'remote use' command. The active remote endpoint will be used for remote builds, key operations, and 'library://' pull and push. You can also 'remote logout' from and 'remote remove' an endpoint that is no longer required.

The remote configuration is stored in $HOME/.singularity/remotes.yaml by default.

Examples

```
All group commands have their own help output:

  $ singularity help remote list
  $ singularity remote list
```

Options

```
-c, --config string   path to the file holding remote endpoint configurations (default "/
↪home/circleci/.singularity/remote.yaml")
-h, --help            help for remote
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity remote add* - Add a new singularity remote endpoint * *singularity remote get-login-password* - Retrieves the cli secret for the currently logged in user * *singularity remote list* - List all singularity remote endpoints that are configured * *singularity remote login* - Login to a singularity remote endpoint * *singularity remote logout* - Log out from a singularity remote endpoint * *singularity remote remove* - Remove an existing singularity remote endpoint * *singularity remote status* - Check the status of the singularity services at an endpoint, and your authentication token * *singularity remote use* - Set a singularity remote endpoint to be actively used

Auto generated by spf13/cobra on 23-Nov-2023

8.2.69 singularity remote add

Add a new singularity remote endpoint

Synopsis

The 'remote add' command allows you to add a new remote endpoint to be used for singularity remote services. Authentication with a newly created endpoint will occur automatically.

```
singularity remote add [add options...] <remote_name> <remote_URI>
```

Examples

```
$ singularity remote add SylabsCloud cloud.sylabs.io
```

Options

```
-g, --global              edit the list of globally configured remote endpoints
-h, --help                help for add
-i, --insecure            allow connection to an insecure http remote
-n, --no-default          do not designate the newly-added remote endpoint as the default
    --no-login            skip automatic login step
    --tokenfile string    path to the file holding auth token for login (remote endpoints␣
↪only)
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.70 singularity remote get-login-password

Retrieves the cli secret for the currently logged in user

Synopsis

The 'remote get-login-password' command allows you to retrieve the cli secret for the currently user.

```
singularity remote get-login-password
```

Examples

```
$ singularity remote get-login-password | docker login -u user --password-stdin https://
↪harbor.sylabs.io
```

Options

```
-h, --help   help for get-login-password
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.71 singularity remote list

List all singularity remote endpoints that are configured

Synopsis

The 'remote list' command lists all remote endpoints configured for use.

The current remote is indicated by 'YES' in the 'ACTIVE' column and can be changed with the 'remote use' command.

```
singularity remote list
```

Examples

```
$ singularity remote list
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.72 singularity remote login

Login to a singularity remote endpoint

Synopsis

The 'remote login' command allows you to set credentials for a specific endpoint.

If no endpoint is specified, the command will login to the currently active remote endpoint. This is cloud.sylabs.io by default.

```
singularity remote login [login options...] <remote_name>
```

Examples

```
To log in to an endpoint:
$ singularity remote login SylabsCloud
```

Options

```
-h, --help               help for login
-i, --insecure           allow insecure login
-p, --password string    password / token to authenticate with
    --password-stdin     take password from standard input
    --tokenfile string   path to the file holding auth token for login (remote endpoints␣
↪only)
-u, --username string    username to authenticate with
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.73 singularity remote logout

Log out from a singularity remote endpoint

Synopsis

The 'remote logout' command allows you to log out from a singularity specific endpoint. If no endpoint or service is specified, it will logout from the currently active remote endpoint.

```
singularity remote logout <remote_name>
```

Examples

```
To log out from an endpoint
$ singularity remote logout SylabsCloud
```

Options

```
-h, --help   help for logout
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.74 singularity remote remove

Remove an existing singularity remote endpoint

Synopsis

The 'remote remove' command allows you to remove an existing remote endpoint from the list of potential endpoints to use.

```
singularity remote remove [remove options...] <remote_name>
```

Examples

```
$ singularity remote remove SylabsCloud
```

Options

```
-g, --global   edit the list of globally configured remote endpoints
-h, --help     help for remove
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.75 singularity remote status

Check the status of the singularity services at an endpoint, and your authentication token

Synopsis

The 'remote status' command checks the status of the specified remote endpoint and reports the availability of services and their versions, and reports the user's logged-in status (or lack thereof) on that endpoint. If no endpoint is specified, it will check the status of the default remote (SylabsCloud). If you have logged in with an authentication token the validity of that token will be checked.

```
singularity remote status [remote_name]
```

Examples

```
$ singularity remote status SylabsCloud
```

Options

```
-h, --help   help for status
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.76 singularity remote use

Set a singularity remote endpoint to be actively used

Synopsis

The 'remote use' command sets the remote to be used by default by any command that interacts with Singularity services.

```
singularity remote use [use options...] <remote_name>
```

Examples

```
$ singularity remote use SylabsCloud
```

Options

```
-e, --exclusive   set the endpoint as exclusive (root user only, imply --global)
-g, --global      edit the list of globally configured remote endpoints
-h, --help        help for use
```

SEE ALSO

- *singularity remote* - Manage singularity remote endpoints

Auto generated by spf13/cobra on 23-Nov-2023

8.2.77 singularity run

Run the user-defined default command within a container

Synopsis

This command will launch a Singularity container and execute a runscript if one is defined for that container. The runscript is a metadata file within the container that contains shell commands. If the file is present (and executable) then this command will execute that file within the container automatically. All arguments following the container name will be passed directly to the runscript.

singularity run accepts the following container formats:

*.sif Singularity Image Format (SIF). Native to Singularity 3.0+

*.sqsh SquashFS format. Native to Singularity 2.4+

*.img ext3 format. Native to Singularity versions < 2.4.

directory/ sandbox format. Directory containing a valid root file
system and optionally Singularity meta-data.

instance://* A local running instance of a container. (See the instance
command group.)

library://* A SIF container hosted on a Library
(default https://cloud.sylabs.io/library)

docker://* A Docker/OCI container hosted on Docker Hub or another
OCI registry.

shub://* A container hosted on Singularity Hub.

oras://* A SIF container hosted on an OCI registry that supports
the OCI Registry As Storage (ORAS) specification.

```
singularity run [run options...] <container>
```

Examples

```
# Here we see that the runscript prints "Hello world: "
$ singularity exec /tmp/debian.sif cat /singularity
#!/bin/sh
echo "Hello world: "

# It runs with our inputs when we run the image
$ singularity run /tmp/debian.sif one two three
Hello world: one two three

# Note that this does the same thing
$ ./tmp/debian.sif one two three
```

Options

```
    --add-caps string               a comma separated capability list to add
    --allow-setuid                  allow setuid binaries in container (root only)
    --app string                    set an application to run inside a container
    --apply-cgroups string          apply cgroups from file for container processes␣
↪(root only)
-B, --bind strings                  a user-bind path specification. spec has the format␣
↪src[:dest[:opts]], where src and dest are outside and inside paths. If dest is not␣
↪given, it is set equal to src. Mount options ('opts') may be specified as 'ro' (read-
↪only) or 'rw' (read/write, which is the default). Multiple bind paths can be given by␣
↪a comma separated list.
    --blkio-weight int              Block IO relative weight in range 10-1000, 0 to␣
↪disable
    --blkio-weight-device strings   Device specific block IO relative weight
    --cdi-dirs strings              comma-separated list of directories in which CDI␣
↪should look for device definition JSON files. If omitted, default will be: /etc/cdi,/
↪var/run/cdi
-e, --cleanenv                      clean environment before running container
    --compat                        apply settings for increased OCI/Docker␣
↪compatibility. Infers --containall, --no-init, --no-umask, --no-eval, --writable-tmpfs.
-c, --contain                       use minimal /dev and empty other directories (e.g. /
↪tmp and $HOME) instead of sharing filesystems from your host
-C, --containall                    contain not only file systems, but also PID, IPC,␣
↪and environment
    --cpu-shares int                CPU shares for container (default -1)
    --cpus string                   Number of CPUs available to container
    --cpuset-cpus string            List of host CPUs available to container
    --cpuset-mems string            List of host memory nodes available to container
    --cwd string                    initial working directory for payload process inside␣
↪the container (synonym for --pwd)
    --device strings                fully-qualified CDI device name(s). A fully-
↪qualified CDI device name consists of a VENDOR, CLASS, and NAME, which are combined as␣
↪follows: <VENDOR>/<CLASS>=<NAME> (e.g. vendor.com/device=mydevice). Multiple fully-
↪qualified CDI device names can be given as a comma separated list.
    --disable-cache                 dont use cache, and dont create cache
    --dns string                    list of DNS server separated by commas to add in␣
↪resolv.conf
    --docker-host string            specify a custom Docker daemon host
    --docker-login                  login to a Docker Repository interactively
    --drop-caps string              a comma separated capability list to drop
    --env stringToString            pass environment variable to contained process␣
↪(default [])
    --env-file string               pass environment variables from file to contained␣
↪process
-f, --fakeroot                      run container in new user namespace as uid 0
    --fusemount strings             A FUSE filesystem mount specification of the form '
↪<type>:<fuse command> <mountpoint>' - where <type> is 'container' or 'host',␣
↪specifying where the mount will be performed ('container-daemon' or 'host-daemon' will␣
↪run the FUSE process detached). <fuse command> is the path to the FUSE executable,␣
↪plus options for the mount. <mountpoint> is the location in the container to which the␣
↪FUSE mount will be attached. E.g. 'container:sshfs 10.0.0.1:/ /sshfs'. Implies --pid.
-h, --help                          help for run
```

(continues on next page)

(continued from previous page)

```
-H, --home string                    a home directory specification. spec can either be a␣
↪src path or src:dest pair. src is the source path of the home directory outside the␣
↪container and dest overrides the home directory within the container. (default "/home/
↪circleci")
    --hostname string                set container hostname. Infers --uts.
-i, --ipc                            run container in a new IPC namespace
    --keep-privs                     let root user keep privileges in container (root␣
↪only)
    --memory string                  Memory limit in bytes
    --memory-reservation string      Memory soft limit in bytes
    --memory-swap string             Swap limit, use -1 for unlimited swap
    --mount stringArray              a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
-n, --net                            run container in a new network namespace (sets up a␣
↪bridge network interface by default)
    --network string                 specify desired network type separated by commas,␣
↪each network will bring up a dedicated interface inside container (default "bridge")
    --network-args strings           specify network arguments to pass to CNI plugins
    --no-compat                      (--oci mode) do not apply settings for increased OCI/
↪Docker compatibility. Emulate native runtime defaults without --contain etc.
    --no-eval                        do not shell evaluate env vars or OCI container CMD/
↪ENTRYPOINT/ARGS
    --no-home                        do NOT mount users home directory if /home is not␣
↪the current working directory
    --no-https                       use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-init                        do NOT start shim process with --pid
    --no-mount strings               disable one or more 'mount xxx' options set in␣
↪singularity.conf, specify absolute destination path to disable a bind path entry, or
↪'bind-paths' to disable all bind path entries.
    --no-oci                         Launch container with native runtime
    --no-pid                         do not run container in a new PID namespace
    --no-privs                       drop all privileges in container (root only in non-
↪OCI mode)
    --no-setgroups                   disable setgroups when entering --fakeroot user␣
↪namespace
    --no-tmp-sandbox                 Prohibits unpacking of images into temporary sandbox␣
↪dirs
    --no-umask                       do not propagate umask to the container, set default␣
↪0022 umask
    --nv                             enable Nvidia support
    --nvccli                         use nvidia-container-cli for GPU setup (experimental)
    --oci                            Launch container with OCI runtime (experimental)
    --oom-kill-disable               Disable OOM killer
-o, --overlay strings                use an overlayFS image for persistent data storage␣
↪or as read-only layer of container
    --passphrase                     prompt for an encryption passphrase
    --pem-path string                enter an path to a PEM formatted RSA key for an␣
↪encrypted container
-p, --pid                            run container in a new PID namespace
    --pids-limit int                 Limit number of container PIDs, use -1 for unlimited
    --rocm                           enable experimental Rocm support
```

(continues on next page)

(continued from previous page)

```
-S, --scratch strings               include a scratch directory within the container␣
↪that is linked to a temporary dir (use -W to force location)
    --security strings              enable security features (SELinux, Apparmor, Seccomp)
    --sif-fuse                      attempt FUSE mount of SIF (unprivileged / user␣
↪namespace only) (experimental)
-u, --userns                        run container in a new user namespace, allowing␣
↪Singularity to run completely unprivileged on recent kernels. This disables some␣
↪features of Singularity, for example it only works with sandbox images.
    --uts                           run container in a new UTS namespace
-W, --workdir string                working directory to be used for /tmp and /var/tmp␣
↪(if -c/--contain was also used)
-w, --writable                      by default all Singularity containers are available␣
↪as read only. This option makes the file system accessible as read/write.
    --writable-tmpfs                makes the file system accessible as read-write with␣
↪non persistent data (with overlay support only)
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.78 singularity run-help

Show the user-defined help for an image

Synopsis

The help text is from the '%help' section of the definition file. If you are using the '–apps' option, the help text is instead from that app's '%apphelp' section.

```
singularity run-help <image path>
```

Examples

```
$ cat my_container.def
Bootstrap: docker
From: busybox

%help
    Some help for this container

%apphelp foo
    Some help for application 'foo' in this container

$ sudo singularity build my_container.sif my_container.def
```

(continues on next page)

(continued from previous page)

```
Using container recipe deffile: my_container.def
[...snip...]
Cleaning up...

$ singularity run-help my_container.sif

  Some help for this container

$ singularity run-help --app foo my_container.sif

  Some help for application in this container
```

Options

```
    --app string   show the help for an app
-h, --help         help for run-help
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.79 singularity search

Search a Container Library for images

Synopsis

Search a Container Library for container images matching the search query. (default cloud.sylabs.io). You can specify an alternate architecture, and/or limit the results to only signed images.

```
singularity search [search options...] <search_query>
```

Examples

```
$ singularity search lolcow
$ singularity search --arch arm64 alpine
$ singularity search --signed tensorflow
```

Options

```
    --arch string      architecture to search for (default "amd64")
-h, --help             help for search
    --library string   URI for library to search
    --signed           architecture to search for
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.80 singularity shell

Run a shell within a container

Synopsis

singularity shell supports the following formats:

*.sif Singularity Image Format (SIF). Native to Singularity 3.0+

*.sqsh SquashFS format. Native to Singularity 2.4+

*.img ext3 format. Native to Singularity versions < 2.4.

directory/ sandbox format. Directory containing a valid root file
system and optionally Singularity meta-data.

instance://* A local running instance of a container. (See the instance
command group.)

library://* A SIF container hosted on a Library
(default https://cloud.sylabs.io/library)

docker://* A Docker/OCI container hosted on Docker Hub or another
OCI registry.

shub://* A container hosted on Singularity Hub.

oras://* A SIF container hosted on an OCI registry that supports
the OCI Registry As Storage (ORAS) specification.

```
singularity shell [shell options...] <container>
```

Examples

```
$ singularity shell /tmp/Debian.sif
Singularity/Debian.sif> pwd
/home/gmk/test
Singularity/Debian.sif> exit

$ singularity shell -C /tmp/Debian.sif
Singularity/Debian.sif> pwd
/home/gmk
Singularity/Debian.sif> ls -l
total 0
Singularity/Debian.sif> exit

$ sudo singularity shell -w /tmp/Debian.sif
$ sudo singularity shell --writable /tmp/Debian.sif

$ singularity shell instance://my_instance

$ singularity shell instance://my_instance
Singularity: Invoking an interactive shell within container...
Singularity container:~> ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
ubuntu       1     0  0 20:00 ?        00:00:00 /usr/local/bin/singularity/bin/sinit
ubuntu       2     0  0 20:01 pts/8    00:00:00 /bin/bash --norc
ubuntu       3     2  0 20:02 pts/8    00:00:00 ps -ef
```

Options

```
    --add-caps string               a comma separated capability list to add
    --allow-setuid                  allow setuid binaries in container (root only)
    --app string                    set an application to run inside a container
    --apply-cgroups string          apply cgroups from file for container processes␣
↪(root only)
-B, --bind strings                  a user-bind path specification. spec has the format␣
↪src[:dest[:opts]], where src and dest are outside and inside paths. If dest is not␣
↪given, it is set equal to src. Mount options ('opts') may be specified as 'ro' (read-
↪only) or 'rw' (read/write, which is the default). Multiple bind paths can be given by␣
↪a comma separated list.
    --blkio-weight int              Block IO relative weight in range 10-1000, 0 to␣
↪disable
    --blkio-weight-device strings   Device specific block IO relative weight
    --cdi-dirs strings              comma-separated list of directories in which CDI␣
↪should look for device definition JSON files. If omitted, default will be: /etc/cdi,/
↪var/run/cdi
-e, --cleanenv                      clean environment before running container
    --compat                        apply settings for increased OCI/Docker␣
↪compatibility. Infers --containall, --no-init, --no-umask, --no-eval, --writable-tmpfs.
-c, --contain                       use minimal /dev and empty other directories (e.g. /
↪tmp and $HOME) instead of sharing filesystems from your host
-C, --containall                    contain not only file systems, but also PID, IPC,␣
```

(continues on next page)

(continued from previous page)

```
↪and environment
    --cpu-shares int                CPU shares for container (default -1)
    --cpus string                   Number of CPUs available to container
    --cpuset-cpus string            List of host CPUs available to container
    --cpuset-mems string            List of host memory nodes available to container
    --cwd string                    initial working directory for payload process inside␣
↪the container (synonym for --pwd)
    --device strings                fully-qualified CDI device name(s). A fully-
↪qualified CDI device name consists of a VENDOR, CLASS, and NAME, which are combined as␣
↪follows: <VENDOR>/<CLASS>=<NAME> (e.g. vendor.com/device=mydevice). Multiple fully-
↪qualified CDI device names can be given as a comma separated list.
    --disable-cache                 dont use cache, and dont create cache
    --dns string                    list of DNS server separated by commas to add in␣
↪resolv.conf
    --docker-host string            specify a custom Docker daemon host
    --docker-login                  login to a Docker Repository interactively
    --drop-caps string              a comma separated capability list to drop
    --env stringToString            pass environment variable to contained process␣
↪(default [])
    --env-file string               pass environment variables from file to contained␣
↪process
-f, --fakeroot                      run container in new user namespace as uid 0
    --fusemount strings             A FUSE filesystem mount specification of the form '
↪<type>:<fuse command> <mountpoint>' - where <type> is 'container' or 'host',␣
↪specifying where the mount will be performed ('container-daemon' or 'host-daemon' will␣
↪run the FUSE process detached). <fuse command> is the path to the FUSE executable,␣
↪plus options for the mount. <mountpoint> is the location in the container to which the␣
↪FUSE mount will be attached. E.g. 'container:sshfs 10.0.0.1:/ /sshfs'. Implies --pid.
-h, --help                          help for shell
-H, --home string                   a home directory specification. spec can either be a␣
↪src path or src:dest pair. src is the source path of the home directory outside the␣
↪container and dest overrides the home directory within the container. (default "/home/
↪circleci")
    --hostname string               set container hostname. Infers --uts.
-i, --ipc                           run container in a new IPC namespace
    --keep-privs                    let root user keep privileges in container (root␣
↪only)
    --memory string                 Memory limit in bytes
    --memory-reservation string     Memory soft limit in bytes
    --memory-swap string            Swap limit, use -1 for unlimited swap
    --mount stringArray             a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
-n, --net                           run container in a new network namespace (sets up a␣
↪bridge network interface by default)
    --network string                specify desired network type separated by commas,␣
↪each network will bring up a dedicated interface inside container (default "bridge")
    --network-args strings          specify network arguments to pass to CNI plugins
    --no-compat                     (--oci mode) do not apply settings for increased OCI/
↪Docker compatibility. Emulate native runtime defaults without --contain etc.
    --no-eval                       do not shell evaluate env vars or OCI container CMD/
↪ENTRYPOINT/ARGS
    --no-home                       do NOT mount users home directory if /home is not␣
```

(continues on next page)

(continued from previous page)

```
↪the current working directory
    --no-https                      use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-init                       do NOT start shim process with --pid
    --no-mount strings              disable one or more 'mount xxx' options set in␣
↪singularity.conf, specify absolute destination path to disable a bind path entry, or
↪'bind-paths' to disable all bind path entries.
    --no-oci                        Launch container with native runtime
    --no-pid                        do not run container in a new PID namespace
    --no-privs                      drop all privileges in container (root only in non-
↪OCI mode)
    --no-setgroups                  disable setgroups when entering --fakeroot user␣
↪namespace
    --no-tmp-sandbox                Prohibits unpacking of images into temporary sandbox␣
↪dirs
    --no-umask                      do not propagate umask to the container, set default␣
↪0022 umask
    --nv                            enable Nvidia support
    --nvccli                        use nvidia-container-cli for GPU setup (experimental)
    --oci                           Launch container with OCI runtime (experimental)
    --oom-kill-disable              Disable OOM killer
-o, --overlay strings               use an overlayFS image for persistent data storage␣
↪or as read-only layer of container
    --passphrase                    prompt for an encryption passphrase
    --pem-path string               enter an path to a PEM formatted RSA key for an␣
↪encrypted container
-p, --pid                           run container in a new PID namespace
    --pids-limit int                Limit number of container PIDs, use -1 for unlimited
    --rocm                          enable experimental Rocm support
-S, --scratch strings               include a scratch directory within the container␣
↪that is linked to a temporary dir (use -W to force location)
    --security strings              enable security features (SELinux, Apparmor, Seccomp)
-s, --shell string                  path to program to use for interactive shell
    --sif-fuse                      attempt FUSE mount of SIF (unprivileged / user␣
↪namespace only) (experimental)
-u, --userns                        run container in a new user namespace, allowing␣
↪Singularity to run completely unprivileged on recent kernels. This disables some␣
↪features of Singularity, for example it only works with sandbox images.
    --uts                           run container in a new UTS namespace
-W, --workdir string                working directory to be used for /tmp and /var/tmp␣
↪(if -c/--contain was also used)
-w, --writable                      by default all Singularity containers are available␣
↪as read only. This option makes the file system accessible as read/write.
    --writable-tmpfs                makes the file system accessible as read-write with␣
↪non persistent data (with overlay support only)
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.81 singularity sif

Manipulate Singularity Image Format (SIF) images

Synopsis

A set of commands are provided to display elements such as the SIF global header, the data object descriptors and to dump data objects. It is also possible to modify a SIF file via this tool via the add/del commands.

Examples

```
All sif commands have their own help output:

$ singularity help sif list
$ singularity sif list --help
```

Options

```
-h, --help   help for sif
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC) * *singularity sif add* - Add data object * *singularity sif del* - Delete data object * *singularity sif dump* - Dump data object * *singularity sif header* - Display global header * *singularity sif info* - Display data object info * *singularity sif list* - List data objects * *singularity sif new* - Create SIF image * *singularity sif setprim* - Set primary system partition

Auto generated by spf13/cobra on 23-Nov-2023

8.2.82 singularity sif add

Add data object

Synopsis

Add a data object to a SIF image.

```
singularity sif add <sif_path> <object_path> [flags]
```

Examples

```
sif add image.sif recipe.def --datatype 1
sif add image.sif rootfs.squashfs --datatype 4 --parttype 1 --partfs 1 --partarch 2
sif add image.sif signature.bin --datatype 5 --signentity␣
↪433FE984155206BD962725E20E8713472A879943 --signhash 1
```

Options

```
    --alignment int       set alignment [default: 4096 with --datatype 4-Partition, 0␣
↪otherwise]
    --datatype int        the type of data to add
                          [NEEDED, no default]:
                            1-Deffile,        2-EnvVar,        3-Labels,
                            4-Partition,      5-Signature,     6-GenericJSON,
                            7-Generic,        8-CryptoMessage, 9-SBOM,
                            10-OCI.RootIndex, 11-OCI.Blob
    --filename string     set logical filename/handle [default: input filename]
    --groupid uint32      set groupid [default: 0]
-h, --help                help for add
    --link uint32         set link pointer [default: 0]
    --partarch int32      the main architecture used (with --datatype 4-Partition)
                          [NEEDED, no default]:
                            1-386,       2-amd64,     3-arm,
                            4-arm64,     5-ppc64,     6-ppc64le,
                            7-mips,      8-mipsle,    9-mips64,
                            10-mips64le, 11-s390x,    12-riscv64
    --partfs int32        the filesystem used (with --datatype 4-Partition)
                          [NEEDED, no default]:
                            1-Squash,    2-Ext3,      3-ImmuObj,
                            4-Raw
    --parttype int32      the type of partition (with --datatype 4-Partition)
                          [NEEDED, no default]:
                            1-System,    2-PrimSys,   3-Data,
                            4-Overlay
    --sbomformat string   the SBOM format (with --datatype 9-sbom):
                            cyclonedx-json, cyclonedx-xml,  github-json,
                            spdx-json,      spdx-rdf,       spdx-tag-value,
                            spdx-yaml,      syft-json
    --signentity string   the entity that signs (with --datatype 5-Signature)
```

(continues on next page)

(continued from previous page)

```
                          [NEEDED, no default]:
                            example: 433FE984155206BD962725E20E8713472A879943
    --signhash int32      the signature hash used (with --datatype 5-Signature)
                          [NEEDED, no default]:
                            1-SHA256,      2-SHA384,      3-SHA512,
                            4-BLAKE2s_256, 5-BLAKE2b_256
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.83 singularity sif del

Delete data object

Synopsis

Delete a data object from a SIF image.

```
singularity sif del <id> <sif_path>
```

Examples

```
sif del 1 image.sif
```

Options

```
-h, --help   help for del
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.84 singularity sif dump

Dump data object

Synopsis

Dump a data object from a SIF image.

```
singularity sif dump <id> <sif_path>
```

Examples

```
sif dump 1 image.sif
```

Options

```
-h, --help   help for dump
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.85 singularity sif header

Display global header

Synopsis

Display global header from a SIF image.

```
singularity sif header <sif_path>
```

Examples

```
sif header image.sif
```

Options

```
-h, --help   help for header
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.86 singularity sif info

Display data object info

Synopsis

Display info about a data object from a SIF image.

```
singularity sif info <id> <sif_path>
```

Examples

```
sif info 1 image.sif
```

Options

```
-h, --help   help for info
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.87 singularity sif list

List data objects

Synopsis

List data objects from a SIF image.

```
singularity sif list <sif_path>
```

Examples

```
sif list image.sif
```

Options

```
-h, --help   help for list
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.88 singularity sif new

Create SIF image

Synopsis

Create a new, empty SIF image.

```
singularity sif new <sif_path>
```

Examples

```
sif new image.sif
```

Options

```
-h, --help   help for new
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.89 singularity sif setprim

Set primary system partition

Synopsis

Set the primary system partition in a SIF image.

```
singularity sif setprim <id> <sif_path>
```

Examples

```
sif setprim 1 image.sif
```

Options

```
-h, --help   help for setprim
```

SEE ALSO

- *singularity sif* - Manipulate Singularity Image Format (SIF) images

Auto generated by spf13/cobra on 23-Nov-2023

8.2.90 singularity sign

Add digital signature(s) to an image

Synopsis

The sign command allows a user to add one or more digital signatures to a SIF image. By default, one digital signature is added for each object group in the file.

Key material can be provided via PEM-encoded file, or an entity in the PGP keyring. To manage the PGP keyring, see 'singularity help key'.

```
singularity sign [sign options...] <image path>
```

Examples

```
Sign with a private key:
$ singularity sign --key private.pem container.sif

Sign with PGP:
$ singularity sign container.sif
```

Options

```
-g, --group-id uint32   sign objects with the specified group ID
-h, --help              help for sign
    --key string        path to the private key file
-k, --keyidx int        PGP private key to use (index from 'key list --secret')
-i, --sif-id uint32     sign object with the specified ID
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.91 singularity test

Run the user-defined tests within a container

Synopsis

The 'test' command allows you to execute a testscript (if available) inside of a given container

NOTE:
For instances if there is a daemon process running inside the container, then subsequent container commands will all run within the same namespaces. This means that the –writable and –contain options will not be honored as the namespaces have already been configured by the 'singularity start' command.

```
singularity test [exec options...] <image path>
```

Examples

```
Set the '%test' section with a definition file like so:
%test
    echo "hello from test" "$@"

$ singularity test /tmp/debian.sif command
    hello from test command

For additional help, please visit our public documentation pages which are
found at:

    https://www.sylabs.io/docs/
```

Options

```
    --add-caps string               a comma separated capability list to add
    --allow-setuid                  allow setuid binaries in container (root only)
    --app string                    set an application to run inside a container
    --apply-cgroups string          apply cgroups from file for container processes␣
↪(root only)
-B, --bind strings                  a user-bind path specification. spec has the format␣
↪src[:dest[:opts]], where src and dest are outside and inside paths. If dest is not␣
↪given, it is set equal to src. Mount options ('opts') may be specified as 'ro' (read-
↪only) or 'rw' (read/write, which is the default). Multiple bind paths can be given by␣
↪a comma separated list.
    --blkio-weight int              Block IO relative weight in range 10-1000, 0 to␣
↪disable
    --blkio-weight-device strings   Device specific block IO relative weight
    --cdi-dirs strings              comma-separated list of directories in which CDI␣
↪should look for device definition JSON files. If omitted, default will be: /etc/cdi,/
↪var/run/cdi
-e, --cleanenv                      clean environment before running container
    --compat                        apply settings for increased OCI/Docker␣
↪compatibility. Infers --containall, --no-init, --no-umask, --no-eval, --writable-tmpfs.
-c, --contain                       use minimal /dev and empty other directories (e.g. /
↪tmp and $HOME) instead of sharing filesystems from your host
-C, --containall                    contain not only file systems, but also PID, IPC,␣
↪and environment
    --cpu-shares int                CPU shares for container (default -1)
    --cpus string                   Number of CPUs available to container
    --cpuset-cpus string            List of host CPUs available to container
    --cpuset-mems string            List of host memory nodes available to container
    --cwd string                    initial working directory for payload process inside␣
↪the container (synonym for --pwd)
    --device strings                fully-qualified CDI device name(s). A fully-
↪qualified CDI device name consists of a VENDOR, CLASS, and NAME, which are combined as␣
↪follows: <VENDOR>/<CLASS>=<NAME> (e.g. vendor.com/device=mydevice). Multiple fully-
↪qualified CDI device names can be given as a comma separated list.
    --disable-cache                 dont use cache, and dont create cache
    --dns string                    list of DNS server separated by commas to add in␣
```

(continues on next page)

(continued from previous page)

```
↪resolv.conf
    --docker-host string            specify a custom Docker daemon host
    --docker-login                  login to a Docker Repository interactively
    --drop-caps string              a comma separated capability list to drop
    --env stringToString            pass environment variable to contained process␣
↪(default [])
    --env-file string               pass environment variables from file to contained␣
↪process
-f, --fakeroot                      run container in new user namespace as uid 0
    --fusemount strings             A FUSE filesystem mount specification of the form '
↪<type>:<fuse command> <mountpoint>' - where <type> is 'container' or 'host',␣
↪specifying where the mount will be performed ('container-daemon' or 'host-daemon' will␣
↪run the FUSE process detached). <fuse command> is the path to the FUSE executable,␣
↪plus options for the mount. <mountpoint> is the location in the container to which the␣
↪FUSE mount will be attached. E.g. 'container:sshfs 10.0.0.1:/ /sshfs'. Implies --pid.
-h, --help                          help for test
-H, --home string                   a home directory specification. spec can either be a␣
↪src path or src:dest pair. src is the source path of the home directory outside the␣
↪container and dest overrides the home directory within the container. (default "/home/
↪circleci")
    --hostname string               set container hostname. Infers --uts.
-i, --ipc                           run container in a new IPC namespace
    --keep-privs                    let root user keep privileges in container (root␣
↪only)
    --memory string                 Memory limit in bytes
    --memory-reservation string     Memory soft limit in bytes
    --memory-swap string            Swap limit, use -1 for unlimited swap
    --mount stringArray             a mount specification e.g. 'type=bind,source=/opt,
↪destination=/hostopt'.
-n, --net                           run container in a new network namespace (sets up a␣
↪bridge network interface by default)
    --network string                specify desired network type separated by commas,␣
↪each network will bring up a dedicated interface inside container (default "bridge")
    --network-args strings          specify network arguments to pass to CNI plugins
    --no-compat                     (--oci mode) do not apply settings for increased OCI/
↪Docker compatibility. Emulate native runtime defaults without --contain etc.
    --no-eval                       do not shell evaluate env vars or OCI container CMD/
↪ENTRYPOINT/ARGS
    --no-home                       do NOT mount users home directory if /home is not␣
↪the current working directory
    --no-https                      use http instead of https for docker:// oras:// and␣
↪library://<hostname>/... URIs
    --no-init                       do NOT start shim process with --pid
    --no-mount strings              disable one or more 'mount xxx' options set in␣
↪singularity.conf, specify absolute destination path to disable a bind path entry, or
↪'bind-paths' to disable all bind path entries.
    --no-oci                        Launch container with native runtime
    --no-pid                        do not run container in a new PID namespace
    --no-privs                      drop all privileges in container (root only in non-
↪OCI mode)
    --no-setgroups                  disable setgroups when entering --fakeroot user␣
↪namespace
```

(continues on next page)

(continued from previous page)

```
    --no-tmp-sandbox                Prohibits unpacking of images into temporary sandbox␣
↪dirs
    --no-umask                      do not propagate umask to the container, set default␣
↪0022 umask
    --nv                            enable Nvidia support
    --nvccli                        use nvidia-container-cli for GPU setup (experimental)
    --oci                           Launch container with OCI runtime (experimental)
    --oom-kill-disable              Disable OOM killer
-o, --overlay strings               use an overlayFS image for persistent data storage␣
↪or as read-only layer of container
    --passphrase                    prompt for an encryption passphrase
    --pem-path string               enter an path to a PEM formatted RSA key for an␣
↪encrypted container
-p, --pid                           run container in a new PID namespace
    --pids-limit int                Limit number of container PIDs, use -1 for unlimited
    --rocm                          enable experimental Rocm support
-S, --scratch strings               include a scratch directory within the container␣
↪that is linked to a temporary dir (use -W to force location)
    --security strings              enable security features (SELinux, Apparmor, Seccomp)
    --sif-fuse                      attempt FUSE mount of SIF (unprivileged / user␣
↪namespace only) (experimental)
-u, --userns                        run container in a new user namespace, allowing␣
↪Singularity to run completely unprivileged on recent kernels. This disables some␣
↪features of Singularity, for example it only works with sandbox images.
    --uts                           run container in a new UTS namespace
-W, --workdir string                working directory to be used for /tmp and /var/tmp␣
↪(if -c/--contain was also used)
-w, --writable                      by default all Singularity containers are available␣
↪as read only. This option makes the file system accessible as read/write.
    --writable-tmpfs                makes the file system accessible as read-write with␣
↪non persistent data (with overlay support only)
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.92 singularity verify

Verify digital signature(s) within an image

Synopsis

The verify command allows a user to verify one or more digital signatures within a SIF image.

Key material can be provided via PEM-encoded file, or via the PGP keyring. To manage the PGP keyring, see 'singularity help key'.

```
singularity verify [verify options...] <image path>
```

Examples

```
Verify with a public key:
$ singularity verify --key public.pem container.sif

Verify with PGP:
$ singularity verify container.sif
```

Options

```
-a, --all                                verify all objects
    --certificate string                 path to the certificate
    --certificate-intermediates string   path to pool of intermediate certificates
    --certificate-roots string           path to pool of root certificates
-g, --group-id uint32                    verify objects with the specified group ID
-h, --help                               help for verify
-j, --json                               output json
    --key string                         path to the public key file
    --legacy-insecure                    enable verification of (insecure) legacy␣
↪signatures
-l, --local                              only verify with local key(s) in keyring
    --ocsp-verify                        enable online revocation check for certificates
-i, --sif-id uint32                      verify object with the specified ID
-u, --url string                         specify a URL for a key server
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.2.93 singularity version

Show the version for Singularity

Synopsis

Show the version for Singularity

```
singularity version
```

Options

```
-h, --help   help for version
```

SEE ALSO

- *singularity* -

Linux container platform optimized for High Performance Computing (HPC) and Enterprise Performance Computing (EPC)

Auto generated by spf13/cobra on 23-Nov-2023

8.3 Licenses

8.3.1 Documentation

This documentation is subject to the following 3-clause BSD license:

```
Copyright (c) 2017, SingularityWare, LLC. All rights reserved.
Copyright (c) 2018-2023, Sylabs, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
```

(continues on next page)

(continued from previous page)

```
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

8.3.2 SingularityCE Software

License

SingularityCE is subject to the Licenses detailed below.

Licenses for 3rd party works that have been incorporated into the project can be found in LICENSE_THIRD_PARTY.md.

Licenses for dependencies used by this project can be found in LICENSE_DEPENDENCIES.md.

BSD 3-Clause License

Copyright (c) 2015-2017, Gregory M. Kurtzer. All rights reserved.

Copyright (c) 2016-2017, The Regents of the University of California. All rights reserved.

Copyright (c) 2017, SingularityWare, LLC. All rights reserved.

Copyright (c) 2018-2023, Sylabs, Inc. All rights reserved.

Copyright (c) Contributors to the Apptainer project, established as Apptainer a Series of LF Projects LLC.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

LBNL License

Copyright (c) 2016-2017, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from the U.S. Dept. of Energy). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the University of California, Lawrence Berkeley National Laboratory, U.S. Dept. of Energy nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NOTICE. This Software was developed under funding from the U.S. Department of Energy and the U.S. Government consequently retains certain rights. As such, the U.S. Government has been granted for itself and others acting on its behalf a paid-up, nonexclusive, irrevocable, worldwide license in the Software to reproduce, distribute copies to the public, prepare derivative works, and perform publicly and display publicly, and to permit other to do so.

Third Party Licenses

This project incorporates code from the following projects, under the terms of their licenses detailed below.

github.com/opencontainers/runtime-tools

The source file:

`internal/pkg/runtime/engine/config/oci/generate/generate.go`

Contains code from:

`github.com/opencontainers/runtime-tools/generate/config.go`

```
Copyright 2015 The Linux Foundation.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
```

(continues on next page)

(continued from previous page)

```
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
```

(continues on next page)

(continued from previous page)

```
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
         Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
         stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
```

(continues on next page)

(continued from previous page)

```
        that You distribute, all copyright, patent, trademark, and
        attribution notices from the Source form of the Work,
        excluding those notices that do not pertain to any part of
        the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
        distribution, then any Derivative Works that You distribute must
        include a readable copy of the attribution notices contained
        within such NOTICE file, excluding those notices that do not
        pertain to any part of the Derivative Works, in at least one
        of the following places: within a NOTICE text file distributed
        as part of the Derivative Works; within the Source form or
        documentation, if provided along with the Derivative Works; or,
        within a display generated by the Derivative Works, if and
        wherever such third-party notices normally appear. The contents
        of the NOTICE file are for informational purposes only and
        do not modify the License. You may add Your own attribution
        notices within Derivative Works that You distribute, alongside
        or as an addendum to the NOTICE text from the Work, provided
        that such additional attribution notices cannot be construed
        as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.
```

(continues on next page)

(continued from previous page)

```
8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

Go

The source files:

- pkg/sypgp/testdata_test.go
- internal/pkg/util/user/cgo_lookup_unix.go
- internal/pkg/util/passwdfile/passwdfile_unix.go

Contain code from the Go project.

```
Copyright 2011 The Go Authors. All rights reserved.
Use of this source code is governed by a BSD-style
license that can be found in the LICENSE file.
```

The referenced LICENSE file for the Go project is:

https://github.com/golang/go/blob/master/LICENSE

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/containers/podman

The source files:

- internal/app/singularity/oci_linux.go
- internal/app/singularity/oci_attach_linux.go
- internal/app/singularity/oci_create_linux.go

Contain code from the podman project, under the Apache License, Version 2.0.

github.com/containers/conmon

The source files:

- third_party/conmon/*

Are code from the conmon project, under the Apache License, Version 2.0.

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
```

(continues on next page)

(continued from previous page)

```
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.
```

(continues on next page)

(continued from previous page)

```
4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
```

(continues on next page)

(continued from previous page)

```
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
```

(continues on next page)

(continued from previous page)

```
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/cli

The source files:

- `internal/app/singularity/instance_linux.go`

Contain code from the docker cli project, under the Apache License, Version 2.0.

github.com/google/go-containerregistry

The source files:

- `internal/pkg/client/oci/auth.go`

Contain code from the go-containerregistry project, under the Apache License, Version 2.0.

```
Copyright 2018 Google LLC All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/vasi/squashfuse

The source files:

- `third_party/squashfuse/*`

Are code from the squashfuse project, under the following license:

```
The squashfuse distribution as a whole is copyright Dave
Vasilevsky and is subject to the copyright notice reproduced at
```

(continues on next page)

(continued from previous page)

```
the bottom of this file.

The file squashfs_fs.h is copyright Phillip Lougher and, with his permission,
subject to the same license.


Copyright (c) 2012 Dave Vasilevsky <dave@vasilevsky.ca>
                   Phillip Lougher <phillip@squashfs.org.uk>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR(S) ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/containerd/containerd

The source files:

- `internal/pkg/ociplatform/cpuinfo.go`
- `internal/pkg/ociplatform/cpuinfo_linux.go`
- `internal/pkg/ociplatform/cpuinfo_linux_test.go`
- `internal/pkg/ociplatform/database.go`

Contain code from the docker cli project, under the Apache License, Version 2.0.

```
/*
   Copyright The containerd Authors.

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.
*/
```

Dependency Licenses

This project uses a number of dependencies, in accordance with their own license terms. These dependencies are managed via the project `go.mod` and `go.sum` files, and included in a `vendor/` directory in our official source tarballs.

A full build or package of SingularityCE uses all dependencies listed below. If you `import "github.com/sylabs/singularity/v4"` into your own project then you may use a subset of them.

The dependencies and their licenses are as follows:

github.com/BurntSushi/toml

License: MIT

```
The MIT License (MIT)

Copyright (c) 2013 TOML authors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/Netflix/go-expect

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2018 Netflix, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/ProtonMail/go-crypto

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/VividCortex/ewma

License: MIT

```
The MIT License

Copyright (c) 2013 VividCortex

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
```

(continues on next page)

(continued from previous page)

```
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/acarl005/stripansi

License: MIT

```
MIT License

Copyright (c) 2018 Andrew Carlson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/adigunhammedolalekan/registry-auth

License: MIT

```
MIT License

Copyright (c) 2020 Lekan Adigun

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

(continues on next page)

(continued from previous page)

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/apex/log

License: MIT

```
(The MIT License)

Copyright (c) 2015 TJ Holowaychuk tj@tjholowaychuk.com

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/asaskevich/govalidator

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014-2020 Alex Saskevich

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
```

(continues on next page)

(continued from previous page)

```
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/astromechza/etcpwdparse

License: MIT

```
MIT License

Copyright (c) 2017 AstromechZA

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/beorn7/perks/quantile

License: MIT

```
Copyright (C) 2013 Blake Mizerany

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
```

(continues on next page)

(continued from previous page)

```
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/blang/semver/v4

License: MIT

```
The MIT License

Copyright (c) 2014 Benedikt Lang <github at benediktlang.de>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/buger/goterm

License: MIT

```
MIT License

Copyright (c) 2016 Leonid Bugaev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/buger/jsonparser

License: MIT

```
MIT License

Copyright (c) 2016 Leonid Bugaev

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/cespare/xxhash/v2

License: MIT

```
Copyright (c) 2016 Caleb Spare

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/cilium/ebpf

License: MIT

```
MIT License

Copyright (c) 2017 Nathan Sweet
Copyright (c) 2018, 2019 Cloudflare
Copyright (c) 2019 Authors of Cilium

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
```

(continues on next page)

(continued from previous page)

```
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/cloudflare/circl

License: BSD-3-Clause

```
Copyright (c) 2019 Cloudflare. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Cloudflare nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

================================================================================

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
```

(continues on next page)

(continued from previous page)

```
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/container-orchestrated-devices/container-device-interface

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.
```

(continues on next page)

(continued from previous page)

```
   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
```

(continues on next page)

(continued from previous page)

```
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.
```

(continues on next page)

(continued from previous page)

```
6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
```

(continues on next page)

(continued from previous page)

```
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containerd/containerd/pkg/userns

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.
```

(continues on next page)

(continued from previous page)

```
   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
```

(continues on next page)

(continued from previous page)

```
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
```

(continues on next page)

(continued from previous page)

```
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright The containerd Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containerd/stargz-snapshotter/estargz

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containernetworking/cni

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containernetworking/plugins

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containers/common

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containers/image/v5

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containers/libtrust

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.
```

(continues on next page)

(continued from previous page)

```
   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
```

(continues on next page)

(continued from previous page)

```
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
```

(continues on next page)

(continued from previous page)

```
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
```

(continues on next page)

(continued from previous page)

```
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2014 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containers/ocicrypt

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
```

(continues on next page)

(continued from previous page)

```
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
```

(continues on next page)

(continued from previous page)

```
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
```

(continues on next page)

(continued from previous page)

```
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
```

(continues on next page)

(continued from previous page)

```
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/containers/storage

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
```

(continues on next page)

(continued from previous page)

```
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
```

(continues on next page)

(continued from previous page)

```
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.
```

(continues on next page)

(continued from previous page)

```
7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2016 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/coreos/go-iptables/iptables

License: Apache-2.0

```
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities
that control, are controlled by, or are under common control with that entity.
For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by
contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising
permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including
but not limited to software source code, documentation source, and configuration
files.

"Object" form shall mean any form resulting from mechanical transformation or
translation of a Source form, including but not limited to compiled object code,
generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made
available under the License, as indicated by a copyright notice that is included
in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that
is based on (or derived from) the Work and for which the editorial revisions,
annotations, elaborations, or other modifications represent, as a whole, an
original work of authorship. For the purposes of this License, Derivative Works
shall not include works that remain separable from, or merely link (or bind by
name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version
of the Work and any modifications or additions to that Work or Derivative Works
thereof, that is intentionally submitted to Licensor for inclusion in the Work
by the copyright owner or by an individual or Legal Entity authorized to submit
on behalf of the copyright owner. For the purposes of this definition,
"submitted" means any form of electronic, verbal, or written communication sent
```

(continues on next page)

(continued from previous page)

```
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and
issue tracking systems that are managed by, or on behalf of, the Licensor for
the purpose of discussing and improving the Work, but excluding communication
that is conspicuously marked or otherwise designated in writing by the copyright
owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf
of whom a Contribution has been received by Licensor and subsequently
incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable (except as stated in this section) patent license to make, have
made, use, offer to sell, sell, import, and otherwise transfer the Work, where
such license applies only to those patent claims licensable by such Contributor
that are necessarily infringed by their Contribution(s) alone or by combination
of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a
Contribution incorporated within the Work constitutes direct or contributory
patent infringement, then any patent licenses granted to You under this License
for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof
in any medium, with or without modifications, and in Source or Object form,
provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of
this License; and
You must cause any modified files to carry prominent notices stating that You
changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute,
all copyright, patent, trademark, and attribution notices from the Source form
of the Work, excluding those notices that do not pertain to any part of the
Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any
Derivative Works that You distribute must include a readable copy of the
attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the
```

(continues on next page)

(continued from previous page)

```
following places: within a NOTICE text file distributed as part of the
Derivative Works; within the Source form or documentation, if provided along
with the Derivative Works; or, within a display generated by the Derivative
Works, if and wherever such third-party notices normally appear. The contents of
the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that
You distribute, alongside or as an addendum to the NOTICE text from the Work,
provided that such additional attribution notices cannot be construed as
modifying the License.
You may add Your own copyright statement to Your modifications and may provide
additional or different license terms and conditions for use, reproduction, or
distribution of Your modifications, or for any such Derivative Works as a whole,
provided Your use, reproduction, and distribution of the Work otherwise complies
with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted
for inclusion in the Work by You to the Licensor shall be under the terms and
conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding
such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks,
service marks, or product names of the Licensor, except as required for
reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the
Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied,
including, without limitation, any warranties or conditions of TITLE,
NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are
solely responsible for determining the appropriateness of using or
redistributing the Work and assume any risks associated with Your exercise of
permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence),
contract, or otherwise, unless required by applicable law (such as deliberate
and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental,
or consequential damages of any character arising as a result of this License or
out of the use or inability to use the Work (including but not limited to
damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has
```

(continues on next page)

(continued from previous page)

```
been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to
offer, and charge a fee for, acceptance of support, warranty, indemnity, or
other liability obligations and/or rights consistent with this License. However,
in accepting such obligations, You may act only on Your own behalf and on Your
sole responsibility, not on behalf of any other Contributor, and only if You
agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your
accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate
notice, with the fields enclosed by brackets "[]" replaced with your own
identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also
recommend that a file or class name and description of purpose be included on
the same "printed page" as the copyright notice for easier identification within
third-party archives.

   Copyright [yyyy] [name of copyright owner]

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.
```

github.com/coreos/go-systemd/v22/dbus

License: Apache-2.0

```
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
```

(continues on next page)

(continued from previous page)

```
"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities
that control, are controlled by, or are under common control with that entity.
For the purposes of this definition, "control" means (i) the power, direct or
indirect, to cause the direction or management of such entity, whether by
contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising
permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including
but not limited to software source code, documentation source, and configuration
files.

"Object" form shall mean any form resulting from mechanical transformation or
translation of a Source form, including but not limited to compiled object code,
generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made
available under the License, as indicated by a copyright notice that is included
in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that
is based on (or derived from) the Work and for which the editorial revisions,
annotations, elaborations, or other modifications represent, as a whole, an
original work of authorship. For the purposes of this License, Derivative Works
shall not include works that remain separable from, or merely link (or bind by
name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version
of the Work and any modifications or additions to that Work or Derivative Works
thereof, that is intentionally submitted to Licensor for inclusion in the Work
by the copyright owner or by an individual or Legal Entity authorized to submit
on behalf of the copyright owner. For the purposes of this definition,
"submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems, and
issue tracking systems that are managed by, or on behalf of, the Licensor for
the purpose of discussing and improving the Work, but excluding communication
that is conspicuously marked or otherwise designated in writing by the copyright
owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf
of whom a Contribution has been received by Licensor and subsequently
incorporated within the Work.
```

(continues on next page)

(continued from previous page)

```
2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such
Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby
grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable (except as stated in this section) patent license to make, have
made, use, offer to sell, sell, import, and otherwise transfer the Work, where
such license applies only to those patent claims licensable by such Contributor
that are necessarily infringed by their Contribution(s) alone or by combination
of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a
Contribution incorporated within the Work constitutes direct or contributory
patent infringement, then any patent licenses granted to You under this License
for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof
in any medium, with or without modifications, and in Source or Object form,
provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of
this License; and
You must cause any modified files to carry prominent notices stating that You
changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute,
all copyright, patent, trademark, and attribution notices from the Source form
of the Work, excluding those notices that do not pertain to any part of the
Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any
Derivative Works that You distribute must include a readable copy of the
attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the
following places: within a NOTICE text file distributed as part of the
Derivative Works; within the Source form or documentation, if provided along
with the Derivative Works; or, within a display generated by the Derivative
Works, if and wherever such third-party notices normally appear. The contents of
the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that
You distribute, alongside or as an addendum to the NOTICE text from the Work,
provided that such additional attribution notices cannot be construed as
modifying the License.
You may add Your own copyright statement to Your modifications and may provide
additional or different license terms and conditions for use, reproduction, or
```

(continues on next page)

(continued from previous page)

```
distribution of Your modifications, or for any such Derivative Works as a whole,
provided Your use, reproduction, and distribution of the Work otherwise complies
with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted
for inclusion in the Work by You to the Licensor shall be under the terms and
conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding
such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks,
service marks, or product names of the Licensor, except as required for
reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the
Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied,
including, without limitation, any warranties or conditions of TITLE,
NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are
solely responsible for determining the appropriateness of using or
redistributing the Work and assume any risks associated with Your exercise of
permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence),
contract, or otherwise, unless required by applicable law (such as deliberate
and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental,
or consequential damages of any character arising as a result of this License or
out of the use or inability to use the Work (including but not limited to
damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has
been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to
offer, and charge a fee for, acceptance of support, warranty, indemnity, or
other liability obligations and/or rights consistent with this License. However,
in accepting such obligations, You may act only on Your own behalf and on Your
sole responsibility, not on behalf of any other Contributor, and only if You
agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your
```

(continues on next page)

(continued from previous page)

```
accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate
notice, with the fields enclosed by brackets "[]" replaced with your own
identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also
recommend that a file or class name and description of purpose be included on
the same "printed page" as the copyright notice for easier identification within
third-party archives.

   Copyright [yyyy] [name of copyright owner]

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.
```

github.com/cpuguy83/go-md2man/v2/md2man

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014 Brian Goff

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
```

(continues on next page)

(continued from previous page)

```
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/creack/pty

License: MIT

```
Copyright (c) 2011 Keith Rarick

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the
Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute,
sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall
be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS
OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/cyberphone/json-canonicalization/go/src/webpki.org/jsoncanonicalizer

License: Apache-2.0

```
Copyright 2018 Anders Rundgren

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/cyphar/filepath-securejoin

License: BSD-3-Clause

```
Copyright (C) 2014-2015 Docker Inc & Go Authors. All rights reserved.
Copyright (C) 2017 SUSE LLC. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/docker/cli/cli/config

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
```

(continues on next page)

(continued from previous page)

```
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
```

(continues on next page)

(continued from previous page)

```
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
```

(continues on next page)

(continued from previous page)

```
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
```

(continues on next page)

(continued from previous page)

```
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2017 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/distribution

License: Apache-2.0

```
Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

      "You" (or "Your") shall mean an individual or Legal Entity
      exercising permissions granted by this License.

      "Source" form shall mean the preferred form for making modifications,
      including but not limited to software source code, documentation
```

(continues on next page)

(continued from previous page)

```
      source, and configuration files.

      "Object" form shall mean any form resulting from mechanical
      transformation or translation of a Source form, including but
      not limited to compiled object code, generated documentation,
      and conversions to other media types.

      "Work" shall mean the work of authorship, whether in Source or
      Object form, made available under the License, as indicated by a
      copyright notice that is included in or attached to the work
      (an example is provided in the Appendix below).

      "Derivative Works" shall mean any work, whether in Source or Object
      form, that is based on (or derived from) the Work and for which the
      editorial revisions, annotations, elaborations, or other modifications
      represent, as a whole, an original work of authorship. For the purposes
      of this License, Derivative Works shall not include works that remain
      separable from, or merely link (or bind by name) to the interfaces of,
      the Work and Derivative Works thereof.

      "Contribution" shall mean any work of authorship, including
      the original version of the Work and any modifications or additions
      to that Work or Derivative Works thereof, that is intentionally
      submitted to Licensor for inclusion in the Work by the copyright owner
      or by an individual or Legal Entity authorized to submit on behalf of
      the copyright owner. For the purposes of this definition, "submitted"
      means any form of electronic, verbal, or written communication sent
      to the Licensor or its representatives, including but not limited to
      communication on electronic mailing lists, source code control systems,
      and issue tracking systems that are managed by, or on behalf of, the
      Licensor for the purpose of discussing and improving the Work, but
      excluding communication that is conspicuously marked or otherwise
      designated in writing by the copyright owner as "Not a Contribution."

      "Contributor" shall mean Licensor and any individual or Legal Entity
      on behalf of whom a Contribution has been received by Licensor and
      subsequently incorporated within the Work.

   2. Grant of Copyright License. Subject to the terms and conditions of
      this License, each Contributor hereby grants to You a perpetual,
      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
      copyright license to reproduce, prepare Derivative Works of,
      publicly display, publicly perform, sublicense, and distribute the
      Work and such Derivative Works in Source or Object form.

   3. Grant of Patent License. Subject to the terms and conditions of
      this License, each Contributor hereby grants to You a perpetual,
      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
      (except as stated in this section) patent license to make, have made,
      use, offer to sell, sell, import, and otherwise transfer the Work,
      where such license applies only to those patent claims licensable
      by such Contributor that are necessarily infringed by their
```

(continues on next page)

(continued from previous page)

```
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
```

(continues on next page)

(continued from previous page)

```
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
```

(continues on next page)

(continued from previous page)

```
      comment syntax for the file format. We also recommend that a
      file or class name and description of purpose be included on the
      same "printed page" as the copyright notice for easier
      identification within third-party archives.

   Copyright {yyyy} {name of copyright owner}

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.
```

github.com/docker/docker

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.
```

(continues on next page)

(continued from previous page)

```
   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
```

(continues on next page)

(continued from previous page)

```
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
```

(continues on next page)

(continued from previous page)

```
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2018 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/docker-credential-helpers

License: MIT

```
Copyright (c) 2016 David Calavera

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/docker/go-connections

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.
```

(continues on next page)

(continued from previous page)

```
"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
```

(continues on next page)

(continued from previous page)

```
2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
```

(continues on next page)

(continued from previous page)

```
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
```

(continues on next page)

(continued from previous page)

```
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2015 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/go-metrics

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.
```

(continues on next page)

(continued from previous page)

```
   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
```

(continues on next page)

(continued from previous page)

```
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.
```

(continues on next page)

(continued from previous page)

```
5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2016 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
```

(continues on next page)

(continued from previous page)

```
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/go-units

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
```

(continues on next page)

(continued from previous page)

```
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
```

(continues on next page)

(continued from previous page)

```
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.
```

(continues on next page)

(continued from previous page)

```
7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2015 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/docker/libtrust

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2014 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/fatih/color

License: MIT

```
The MIT License (MIT)

Copyright (c) 2013 Fatih Arslan

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
```

(continues on next page)

(continued from previous page)

```
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/felixge/httpsnoop

License: MIT

```
Copyright (c) 2016 Felix Geisendörfer (felix@debuggable.com)

 Permission is hereby granted, free of charge, to any person obtaining a copy
 of this software and associated documentation files (the "Software"), to deal
 in the Software without restriction, including without limitation the rights
 to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in
 all copies or substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 THE SOFTWARE.
```

github.com/fsnotify/fsnotify

License: BSD-3-Clause

```
Copyright © 2012 The Go Authors. All rights reserved.
Copyright © fsnotify Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this
```

(continues on next page)

(continued from previous page)

```
  list of conditions and the following disclaimer in the documentation and/or
  other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used
  to endorse or promote products derived from this software without specific
  prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/garyburd/redigo/internal

License: Unknown

github.com/garyburd/redigo/redis

License: Unknown

github.com/go-jose/go-jose/v3

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
```

(continues on next page)

(continued from previous page)

```
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.
```

(continues on next page)

(continued from previous page)

```
3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
```

(continues on next page)

(continued from previous page)

```
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

(continues on next page)

(continued from previous page)

```
APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-jose/go-jose/v3/json

License: BSD-3-Clause

```
Copyright (c) 2012 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
```

(continues on next page)

(continued from previous page)

```
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/go-log/log

License: MIT

```
Copyright (c) 2017 Go Log

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/go-openapi/analysis

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.
```

(continues on next page)

(continued from previous page)

```
   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
```

(continues on next page)

(continued from previous page)

```
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
```

(continues on next page)

(continued from previous page)

```
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
```

(continues on next page)

(continued from previous page)

```
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/errors

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
```

(continues on next page)

(continued from previous page)

```
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
```

(continues on next page)

(continued from previous page)

```
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
```

(continues on next page)

(continued from previous page)

```
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
```

(continues on next page)

(continued from previous page)

```
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/jsonpointer

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
```

(continues on next page)

(continued from previous page)

```
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
```

(continues on next page)

(continued from previous page)

```
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
```

(continues on next page)

(continued from previous page)

```
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
```

(continues on next page)

(continued from previous page)

```
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/jsonreference

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
```

(continues on next page)

(continued from previous page)

```
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
```

(continues on next page)

(continued from previous page)

```
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.
```

(continues on next page)

(continued from previous page)

```
   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
```

(continues on next page)

(continued from previous page)

```
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/loads

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
```

(continues on next page)

(continued from previous page)

```
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
```

(continues on next page)

(continued from previous page)

```
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.
```

(continues on next page)

(continued from previous page)

```
   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.
```

(continues on next page)

(continued from previous page)

```
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/runtime

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
```

(continues on next page)

(continued from previous page)

```
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.
```

(continues on next page)

(continued from previous page)

```
3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
```

(continues on next page)

(continued from previous page)

```
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.
```

(continues on next page)

(continued from previous page)

```
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/spec

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
```

(continues on next page)

(continued from previous page)

```
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.
```

(continues on next page)

(continued from previous page)

```
3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
```

(continues on next page)

(continued from previous page)

```
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

(continues on next page)

(continued from previous page)

```
APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/strfmt

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.
```

(continues on next page)

(continued from previous page)

```
   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
```

(continues on next page)

(continued from previous page)

```
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
```

(continues on next page)

(continued from previous page)

```
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

(continues on next page)

(continued from previous page)

```
APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/swag

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.
```

(continues on next page)

(continued from previous page)

```
   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
```

(continues on next page)

(continued from previous page)

```
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
```

(continues on next page)

(continued from previous page)

```
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
```

(continues on next page)

(continued from previous page)

```
   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/go-openapi/validate

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
```

(continues on next page)

(continued from previous page)

```
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
```

(continues on next page)

(continued from previous page)

```
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
```

(continues on next page)

(continued from previous page)

```
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
```

(continues on next page)

(continued from previous page)

```
   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/godbus/dbus/v5

License: BSD-2-Clause

```
Copyright (c) 2013, Georg Reinke (<guelfey at gmail dot com>), Google
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
```

(continues on next page)

(continued from previous page)

```
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/gogo/protobuf/proto

License: BSD-3-Clause

```
Copyright (c) 2013, The GoGo Authors. All rights reserved.

Protocol Buffers for Go with Gadgets

Go support for Protocol Buffers - Google's data interchange format

Copyright 2010 The Go Authors.  All rights reserved.
https://github.com/golang/protobuf

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
    * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/golang/protobuf

License: BSD-3-Clause

```
Copyright 2010 The Go Authors.  All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
    * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/google/go-cmp/cmp

License: BSD-3-Clause

```
Copyright (c) 2017 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.
```

(continues on next page)

(continued from previous page)

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/google/go-containerregistry

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
```

(continues on next page)

(continued from previous page)

```
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.
```

(continues on next page)

(continued from previous page)

```
4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
```

(continues on next page)

(continued from previous page)

```
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
```

(continues on next page)

(continued from previous page)

```
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/google/uuid

License: BSD-3-Clause

```
Copyright (c) 2009,2014 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/gorilla/handlers

License: BSD-2-Clause

```
Copyright (c) 2013 The Gorilla Handlers Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/gorilla/mux

License: BSD-3-Clause

```
Copyright (c) 2012-2018 The Gorilla Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

     * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
     * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
     * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
```

(continues on next page)

(continued from previous page)

```
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/gorilla/websocket

License: BSD-2-Clause

```
Copyright (c) 2013 The Gorilla WebSocket Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/gosimple/slug

License: MPL-2.0

```
Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to
    the creation of, or owns Covered Software.

1.2. "Contributor Version"
```

(continues on next page)

(continued from previous page)

```
    means the combination of the Contributions of others (if any) used
    by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.

1.4. "Covered Software"
    means Source Code Form to which the initial Contributor has attached
    the notice in Exhibit A, the Executable Form of such Source Code
    Form, and Modifications of such Source Code Form, in each case
    including portions thereof.

1.5. "Incompatible With Secondary Licenses"
    means

    (a) that the initial Contributor has attached the notice described
        in Exhibit B to the Covered Software; or

    (b) that the Covered Software was made available under the terms of
        version 1.1 or earlier of the License, but not also under the
        terms of a Secondary License.

1.6. "Executable Form"
    means any form of the work other than Source Code Form.

1.7. "Larger Work"
    means a work that combines Covered Software with other material, in
    a separate file or files, that is not Covered Software.

1.8. "License"
    means this document.

1.9. "Licensable"
    means having the right to grant, to the maximum extent possible,
    whether at the time of the initial grant or subsequently, any and
    all of the rights conveyed by this License.

1.10. "Modifications"
    means any of the following:

    (a) any file in Source Code Form that results from an addition to,
        deletion from, or modification of the contents of Covered
        Software; or

    (b) any new file in Source Code Form that contains any Covered
        Software.

1.11. "Patent Claims" of a Contributor
    means any patent claim(s), including without limitation, method,
    process, and apparatus claims, in any patent Licensable by such
    Contributor that would be infringed, but for the grant of the
    License, by the making, using, selling, offering for sale, having
```

(continues on next page)

(continued from previous page)

```
    made, import, or transfer of either its Contributions or its
    Contributor Version.

1.12. "Secondary License"
    means either the GNU General Public License, Version 2.0, the GNU
    Lesser General Public License, Version 2.1, the GNU Affero General
    Public License, Version 3.0, or any later versions of those
    licenses.

1.13. "Source Code Form"
    means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
    means an individual or a legal entity exercising rights under this
    License. For legal entities, "You" includes any entity that
    controls, is controlled by, or is under common control with You. For
    purposes of this definition, "control" means (a) the power, direct
    or indirect, to cause the direction or management of such entity,
    whether by contract or otherwise, or (b) ownership of more than
    fifty percent (50%) of the outstanding shares or beneficial
    ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
    Licensable by such Contributor to use, reproduce, make available,
    modify, display, perform, distribute, and otherwise exploit its
    Contributions, either on an unmodified basis, with Modifications, or
    as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
    for sale, have made, import, and otherwise transfer either its
    Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License.
Notwithstanding Section 2.1(b) above, no patent license is granted by a
```

(continues on next page)

(continued from previous page)

```
Contributor:

(a) for any code that a Contributor has removed from Covered Software;
    or

(b) for infringements caused by: (i) Your and any other third party's
    modifications of Covered Software, or (ii) the combination of its
    Contributions with other software (except as part of its Contributor
    Version); or

(c) under Patent Claims infringed by Covered Software in the absence of
    its Contributions.

This License does not grant any rights in the trademarks, service marks,
or logos of any Contributor (except as may be necessary to comply with
the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to
distribute the Covered Software under a subsequent version of this
License (see Section 10.2) or under the terms of a Secondary License (if
permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its
Contributions are its original creation(s) or it has sufficient rights
to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under
applicable copyright doctrines of fair use, fair dealing, or other
equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted
in Section 2.1.

3. Responsibilities
-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any
Modifications that You create or to which You contribute, must be under
the terms of this License. You must inform recipients that the Source
Code Form of the Covered Software is governed by the terms of this
License, and how they can obtain a copy of this License. You may not
attempt to alter or restrict the recipients' rights in the Source Code
```

(continues on next page)

(continued from previous page)

```
Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code
    Form, as described in Section 3.1, and You must inform recipients of
    the Executable Form how they can obtain a copy of such Source Code
    Form by reasonable means in a timely manner, at a charge no more
    than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this
    License, or sublicense it under different terms, provided that the
    license for the Executable Form does not attempt to limit or alter
    the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice,
provided that You also comply with the requirements of this License for
the Covered Software. If the Larger Work is a combination of Covered
Software with a work governed by one or more Secondary Licenses, and the
Covered Software is not Incompatible With Secondary Licenses, this
License permits You to additionally distribute such Covered Software
under the terms of such Secondary License(s), so that the recipient of
the Larger Work may, at their option, further distribute the Covered
Software under the terms of either this License or such Secondary
License(s).

3.4. Notices

You may not remove or alter the substance of any license notices
(including copyright notices, patent notices, disclaimers of warranty,
or limitations of liability) contained within the Source Code Form of
the Covered Software, except that You may alter any license notices to
the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support,
indemnity or liability obligations to one or more recipients of Covered
Software. However, You may do so only on Your own behalf, and not on
behalf of any Contributor. You must make it absolutely clear that any
such warranty, support, indemnity, or liability obligation is offered by
You alone, and You hereby agree to indemnify every Contributor for any
liability incurred by such Contributor as a result of warranty, support,
indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation
```

(continues on next page)

(continued from previous page)

```
--------------------------------------------------

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Software due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description must
be placed in a text file included with all distributions of the Covered
Software under this License. Except to the extent prohibited by statute
or regulation, such description must be sufficiently detailed for a
recipient of ordinary skill to be able to understand it.

5. Termination
--------------

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
```

(continues on next page)

(continued from previous page)

```
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous
----------------

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License
```

(continues on next page)

(continued from previous page)

```
--------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
of the License under which You originally received the Covered Software,
or under the terms of any subsequent version published by the license
steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a
modified version of this License if you rename the license and remove
any references to the name of the license steward (except to note that
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary
Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

  This Source Code Form is subject to the terms of the Mozilla Public
  License, v. 2.0. If a copy of the MPL was not distributed with this
  file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

  This Source Code Form is "Incompatible With Secondary Licenses", as
  defined by the Mozilla Public License, v. 2.0.
```

github.com/gosimple/unidecode

License: Apache-2.0

```
Copyright 2014 Rainy Cape S.L. <hello@rainycape.com>

Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

      "You" (or "Your") shall mean an individual or Legal Entity
      exercising permissions granted by this License.

      "Source" form shall mean the preferred form for making modifications,
      including but not limited to software source code, documentation
      source, and configuration files.

      "Object" form shall mean any form resulting from mechanical
      transformation or translation of a Source form, including but
      not limited to compiled object code, generated documentation,
      and conversions to other media types.

      "Work" shall mean the work of authorship, whether in Source or
      Object form, made available under the License, as indicated by a
      copyright notice that is included in or attached to the work
      (an example is provided in the Appendix below).

      "Derivative Works" shall mean any work, whether in Source or Object
      form, that is based on (or derived from) the Work and for which the
      editorial revisions, annotations, elaborations, or other modifications
      represent, as a whole, an original work of authorship. For the purposes
      of this License, Derivative Works shall not include works that remain
      separable from, or merely link (or bind by name) to the interfaces of,
      the Work and Derivative Works thereof.
```

(continues on next page)

(continued from previous page)

```
    "Contribution" shall mean any work of authorship, including
    the original version of the Work and any modifications or additions
    to that Work or Derivative Works thereof, that is intentionally
    submitted to Licensor for inclusion in the Work by the copyright owner
    or by an individual or Legal Entity authorized to submit on behalf of
    the copyright owner. For the purposes of this definition, "submitted"
    means any form of electronic, verbal, or written communication sent
    to the Licensor or its representatives, including but not limited to
    communication on electronic mailing lists, source code control systems,
    and issue tracking systems that are managed by, or on behalf of, the
    Licensor for the purpose of discussing and improving the Work, but
    excluding communication that is conspicuously marked or otherwise
    designated in writing by the copyright owner as "Not a Contribution."

    "Contributor" shall mean Licensor and any individual or Legal Entity
    on behalf of whom a Contribution has been received by Licensor and
    subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
    this License, each Contributor hereby grants to You a perpetual,
    worldwide, non-exclusive, no-charge, royalty-free, irrevocable
    copyright license to reproduce, prepare Derivative Works of,
    publicly display, publicly perform, sublicense, and distribute the
    Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
    this License, each Contributor hereby grants to You a perpetual,
    worldwide, non-exclusive, no-charge, royalty-free, irrevocable
    (except as stated in this section) patent license to make, have made,
    use, offer to sell, sell, import, and otherwise transfer the Work,
    where such license applies only to those patent claims licensable
    by such Contributor that are necessarily infringed by their
    Contribution(s) alone or by combination of their Contribution(s)
    with the Work to which such Contribution(s) was submitted. If You
    institute patent litigation against any entity (including a
    cross-claim or counterclaim in a lawsuit) alleging that the Work
    or a Contribution incorporated within the Work constitutes direct
    or contributory patent infringement, then any patent licenses
    granted to You under this License for that Work shall terminate
    as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
    Work or Derivative Works thereof in any medium, with or without
    modifications, and in Source or Object form, provided that You
    meet the following conditions:

    (a) You must give any other recipients of the Work or
        Derivative Works a copy of this License; and

    (b) You must cause any modified files to carry prominent notices
        stating that You changed the files; and
```

(continues on next page)

(continued from previous page)

```
      (c) You must retain, in the Source form of any Derivative Works
          that You distribute, all copyright, patent, trademark, and
          attribution notices from the Source form of the Work,
          excluding those notices that do not pertain to any part of
          the Derivative Works; and

      (d) If the Work includes a "NOTICE" text file as part of its
          distribution, then any Derivative Works that You distribute must
          include a readable copy of the attribution notices contained
          within such NOTICE file, excluding those notices that do not
          pertain to any part of the Derivative Works, in at least one
          of the following places: within a NOTICE text file distributed
          as part of the Derivative Works; within the Source form or
          documentation, if provided along with the Derivative Works; or,
          within a display generated by the Derivative Works, if and
          wherever such third-party notices normally appear. The contents
          of the NOTICE file are for informational purposes only and
          do not modify the License. You may add Your own attribution
          notices within Derivative Works that You distribute, alongside
          or as an addendum to the NOTICE text from the Work, provided
          that such additional attribution notices cannot be construed
          as modifying the License.

      You may add Your own copyright statement to Your modifications and
      may provide additional or different license terms and conditions
      for use, reproduction, or distribution of Your modifications, or
      for any such Derivative Works as a whole, provided Your use,
      reproduction, and distribution of the Work otherwise complies with
      the conditions stated in this License.

   5. Submission of Contributions. Unless You explicitly state otherwise,
      any Contribution intentionally submitted for inclusion in the Work
      by You to the Licensor shall be under the terms and conditions of
      this License, without any additional terms or conditions.
      Notwithstanding the above, nothing herein shall supersede or modify
      the terms of any separate license agreement you may have executed
      with Licensor regarding such Contributions.

   6. Trademarks. This License does not grant permission to use the trade
      names, trademarks, service marks, or product names of the Licensor,
      except as required for reasonable and customary use in describing the
      origin of the Work and reproducing the content of the NOTICE file.

   7. Disclaimer of Warranty. Unless required by applicable law or
      agreed to in writing, Licensor provides the Work (and each
      Contributor provides its Contributions) on an "AS IS" BASIS,
      WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
      implied, including, without limitation, any warranties or conditions
      of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
      PARTICULAR PURPOSE. You are solely responsible for determining the
      appropriateness of using or redistributing the Work and assume any
      risks associated with Your exercise of permissions under this License.
```

(continues on next page)

(continued from previous page)

```
8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/hashicorp/errwrap

License: MPL-2.0

```
Mozilla Public License, version 2.0

1. Definitions

1.1. “Contributor”

     means each individual or legal entity that creates, contributes to the
     creation of, or owns Covered Software.

1.2. “Contributor Version”

     means the combination of the Contributions of others (if any) used by a
     Contributor and that particular Contributor’s Contribution.

1.3. “Contribution”

     means Covered Software of a particular Contributor.

1.4. “Covered Software”

     means Source Code Form to which the initial Contributor has attached the
     notice in Exhibit A, the Executable Form of such Source Code Form, and
     Modifications of such Source Code Form, in each case including portions
     thereof.

1.5. “Incompatible With Secondary Licenses”
     means

     a. that the initial Contributor has attached the notice described in
        Exhibit B to the Covered Software; or

     b. that the Covered Software was made available under the terms of version
        1.1 or earlier of the License, but not also under the terms of a
        Secondary License.

1.6. “Executable Form”

     means any form of the work other than Source Code Form.

1.7. “Larger Work”

     means a work that combines Covered Software with other material, in a separate
     file or files, that is not Covered Software.

1.8. “License”

     means this document.

1.9. “Licensable”
```

(continues on next page)

(continued from previous page)

```
     means having the right to grant, to the maximum extent possible, whether at the
     time of the initial grant or subsequently, any and all of the rights conveyed by
     this License.

1.10. “Modifications”

     means any of the following:

     a. any file in Source Code Form that results from an addition to, deletion
        from, or modification of the contents of Covered Software; or

     b. any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor

      means any patent claim(s), including without limitation, method, process,
      and apparatus claims, in any patent Licensable by such Contributor that
      would be infringed, but for the grant of the License, by the making,
      using, selling, offering for sale, having made, import, or transfer of
      either its Contributions or its Contributor Version.

1.12. “Secondary License”

      means either the GNU General Public License, Version 2.0, the GNU Lesser
      General Public License, Version 2.1, the GNU Affero General Public
      License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form”

      means the form of the work preferred for making modifications.

1.14. “You” (or “Your”)

      means an individual or a legal entity exercising rights under this
      License. For legal entities, “You” includes any entity that controls, is
      controlled by, or is under common control with You. For purposes of this
      definition, “control” means (a) the power, direct or indirect, to cause
      the direction or management of such entity, whether by contract or
      otherwise, or (b) ownership of more than fifty percent (50%) of the
      outstanding shares or beneficial ownership of such entity.


2. License Grants and Conditions

2.1. Grants

     Each Contributor hereby grants You a world-wide, royalty-free,
     non-exclusive license:

     a. under intellectual property rights (other than patent or trademark)
        Licensable by such Contributor to use, reproduce, make available,
```

(continues on next page)

(continued from previous page)

```
        modify, display, perform, distribute, and otherwise exploit its
        Contributions, either on an unmodified basis, with Modifications, or as
        part of a Larger Work; and

     b. under Patent Claims of such Contributor to make, use, sell, offer for
        sale, have made, import, and otherwise transfer either its Contributions
        or its Contributor Version.

2.2. Effective Date

     The licenses granted in Section 2.1 with respect to any Contribution become
     effective for each Contribution on the date the Contributor first distributes
     such Contribution.

2.3. Limitations on Grant Scope

     The licenses granted in this Section 2 are the only rights granted under this
     License. No additional rights or licenses will be implied from the distribution
     or licensing of Covered Software under this License. Notwithstanding Section
     2.1(b) above, no patent license is granted by a Contributor:

     a. for any code that a Contributor has removed from Covered Software; or

     b. for infringements caused by: (i) Your and any other third party's
        modifications of Covered Software, or (ii) the combination of its
        Contributions with other software (except as part of its Contributor
        Version); or

     c. under Patent Claims infringed by Covered Software in the absence of its
        Contributions.

     This License does not grant any rights in the trademarks, service marks, or
     logos of any Contributor (except as may be necessary to comply with the
     notice requirements in Section 3.4).

2.4. Subsequent Licenses

     No Contributor makes additional grants as a result of Your choice to
     distribute the Covered Software under a subsequent version of this License
     (see Section 10.2) or under the terms of a Secondary License (if permitted
     under the terms of Section 3.3).

2.5. Representation

     Each Contributor represents that the Contributor believes its Contributions
     are its original creation(s) or it has sufficient rights to grant the
     rights to its Contributions conveyed by this License.

2.6. Fair Use

     This License is not intended to limit any rights You have under applicable
     copyright doctrines of fair use, fair dealing, or other equivalents.
```

(continues on next page)

(continued from previous page)

```
2.7. Conditions

     Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in
     Section 2.1.


3. Responsibilities

3.1. Distribution of Source Form

     All distribution of Covered Software in Source Code Form, including any
     Modifications that You create or to which You contribute, must be under the
     terms of this License. You must inform recipients that the Source Code Form
     of the Covered Software is governed by the terms of this License, and how
     they can obtain a copy of this License. You may not attempt to alter or
     restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

     If You distribute Covered Software in Executable Form then:

     a. such Covered Software must also be made available in Source Code Form,
        as described in Section 3.1, and You must inform recipients of the
        Executable Form how they can obtain a copy of such Source Code Form by
        reasonable means in a timely manner, at a charge no more than the cost
        of distribution to the recipient; and

     b. You may distribute such Executable Form under the terms of this License,
        or sublicense it under different terms, provided that the license for
        the Executable Form does not attempt to limit or alter the recipients'
        rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

     You may create and distribute a Larger Work under terms of Your choice,
     provided that You also comply with the requirements of this License for the
     Covered Software. If the Larger Work is a combination of Covered Software
     with a work governed by one or more Secondary Licenses, and the Covered
     Software is not Incompatible With Secondary Licenses, this License permits
     You to additionally distribute such Covered Software under the terms of
     such Secondary License(s), so that the recipient of the Larger Work may, at
     their option, further distribute the Covered Software under the terms of
     either this License or such Secondary License(s).

3.4. Notices

     You may not remove or alter the substance of any license notices (including
     copyright notices, patent notices, disclaimers of warranty, or limitations
     of liability) contained within the Source Code Form of the Covered
     Software, except that You may alter any license notices to the extent
     required to remedy known factual inaccuracies.
```

(continues on next page)

(continued from previous page)

```
3.5. Application of Additional Terms

     You may choose to offer, and to charge a fee for, warranty, support,
     indemnity or liability obligations to one or more recipients of Covered
     Software. However, You may do so only on Your own behalf, and not on behalf
     of any Contributor. You must make it absolutely clear that any such
     warranty, support, indemnity, or liability obligation is offered by You
     alone, and You hereby agree to indemnify every Contributor for any
     liability incurred by such Contributor as a result of warranty, support,
     indemnity or liability terms You offer. You may include additional
     disclaimers of warranty and limitations of liability specific to any
     jurisdiction.

4. Inability to Comply Due to Statute or Regulation

   If it is impossible for You to comply with any of the terms of this License
   with respect to some or all of the Covered Software due to statute, judicial
   order, or regulation then You must: (a) comply with the terms of this License
   to the maximum extent possible; and (b) describe the limitations and the code
   they affect. Such description must be placed in a text file included with all
   distributions of the Covered Software under this License. Except to the
   extent prohibited by statute or regulation, such description must be
   sufficiently detailed for a recipient of ordinary skill to be able to
   understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You
     fail to comply with any of its terms. However, if You become compliant,
     then the rights granted under this License from a particular Contributor
     are reinstated (a) provisionally, unless and until such Contributor
     explicitly and finally terminates Your grants, and (b) on an ongoing basis,
     if such Contributor fails to notify You of the non-compliance by some
     reasonable means prior to 60 days after You have come back into compliance.
     Moreover, Your grants from a particular Contributor are reinstated on an
     ongoing basis if such Contributor notifies You of the non-compliance by
     some reasonable means, this is the first time You have received notice of
     non-compliance with this License from such Contributor, and You become
     compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
     infringement claim (excluding declaratory judgment actions, counter-claims,
     and cross-claims) alleging that a Contributor Version directly or
     indirectly infringes any patent, then the rights granted to You by any and
     all Contributors for the Covered Software under Section 2.1 of this License
     shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user
     license agreements (excluding distributors and resellers) which have been
     validly granted by You or Your distributors under this License prior to
     termination shall survive termination.
```

(continues on next page)

(continued from previous page)

```
6. Disclaimer of Warranty

   Covered Software is provided under this License on an “as is” basis, without
   warranty of any kind, either expressed, implied, or statutory, including,
   without limitation, warranties that the Covered Software is free of defects,
   merchantable, fit for a particular purpose or non-infringing. The entire
   risk as to the quality and performance of the Covered Software is with You.
   Should any Covered Software prove defective in any respect, You (not any
   Contributor) assume the cost of any necessary servicing, repair, or
   correction. This disclaimer of warranty constitutes an essential part of this
   License. No use of  any Covered Software is authorized under this License
   except under this disclaimer.

7. Limitation of Liability

   Under no circumstances and under no legal theory, whether tort (including
   negligence), contract, or otherwise, shall any Contributor, or anyone who
   distributes Covered Software as permitted above, be liable to You for any
   direct, indirect, special, incidental, or consequential damages of any
   character including, without limitation, damages for lost profits, loss of
   goodwill, work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses, even if such party shall have been
   informed of the possibility of such damages. This limitation of liability
   shall not apply to liability for death or personal injury resulting from such
   party’s negligence to the extent applicable law prohibits such limitation.
   Some jurisdictions do not allow the exclusion or limitation of incidental or
   consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

   Any litigation relating to this License may be brought only in the courts of
   a jurisdiction where the defendant maintains its principal place of business
   and such litigation shall be governed by laws of that jurisdiction, without
   reference to its conflict-of-law provisions. Nothing in this Section shall
   prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

   This License represents the complete agreement concerning the subject matter
   hereof. If any provision of this License is held to be unenforceable, such
   provision shall be reformed only to the extent necessary to make it
   enforceable. Any law or regulation which provides that the language of a
   contract shall be construed against the drafter shall not be used to construe
   this License against a Contributor.


10. Versions of the License

10.1. New Versions

      Mozilla Foundation is the license steward. Except as provided in Section
```

(continues on next page)

(continued from previous page)

```
      10.3, no one other than the license steward has the right to modify or
      publish new versions of this License. Each version will be given a
      distinguishing version number.

10.2. Effect of New Versions

      You may distribute the Covered Software under the terms of the version of
      the License under which You originally received the Covered Software, or
      under the terms of any subsequent version published by the license
      steward.

10.3. Modified Versions

      If you create software not governed by this License, and you want to
      create a new license for such software, you may create and use a modified
      version of this License if you rename the license and remove any
      references to the name of the license steward (except to note that such
      modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses
      If You choose to distribute Source Code Form that is Incompatible With
      Secondary Licenses under the terms of this version of the License, the
      notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

      This Source Code Form is subject to the
      terms of the Mozilla Public License, v.
      2.0. If a copy of the MPL was not
      distributed with this file, You can
      obtain one at
      http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then
You may include the notice in a location (such as a LICENSE file in a relevant
directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - “Incompatible With Secondary Licenses” Notice

      This Source Code Form is “Incompatible
      With Secondary Licenses”, as defined by
      the Mozilla Public License, v. 2.0.
```

github.com/hashicorp/go-multierror

License: MPL-2.0

```
Mozilla Public License, version 2.0

1. Definitions

1.1. “Contributor”

     means each individual or legal entity that creates, contributes to the
     creation of, or owns Covered Software.

1.2. “Contributor Version”

     means the combination of the Contributions of others (if any) used by a
     Contributor and that particular Contributor’s Contribution.

1.3. “Contribution”

     means Covered Software of a particular Contributor.

1.4. “Covered Software”

     means Source Code Form to which the initial Contributor has attached the
     notice in Exhibit A, the Executable Form of such Source Code Form, and
     Modifications of such Source Code Form, in each case including portions
     thereof.

1.5. “Incompatible With Secondary Licenses”
     means

     a. that the initial Contributor has attached the notice described in
        Exhibit B to the Covered Software; or

     b. that the Covered Software was made available under the terms of version
        1.1 or earlier of the License, but not also under the terms of a
        Secondary License.

1.6. “Executable Form”

     means any form of the work other than Source Code Form.

1.7. “Larger Work”

     means a work that combines Covered Software with other material, in a separate
     file or files, that is not Covered Software.

1.8. “License”

     means this document.

1.9. “Licensable”
```

(continues on next page)

(continued from previous page)

```
     means having the right to grant, to the maximum extent possible, whether at the
     time of the initial grant or subsequently, any and all of the rights conveyed by
     this License.

1.10. “Modifications”

     means any of the following:

     a. any file in Source Code Form that results from an addition to, deletion
        from, or modification of the contents of Covered Software; or

     b. any new file in Source Code Form that contains any Covered Software.

1.11. “Patent Claims” of a Contributor

      means any patent claim(s), including without limitation, method, process,
      and apparatus claims, in any patent Licensable by such Contributor that
      would be infringed, but for the grant of the License, by the making,
      using, selling, offering for sale, having made, import, or transfer of
      either its Contributions or its Contributor Version.

1.12. “Secondary License”

      means either the GNU General Public License, Version 2.0, the GNU Lesser
      General Public License, Version 2.1, the GNU Affero General Public
      License, Version 3.0, or any later versions of those licenses.

1.13. “Source Code Form”

      means the form of the work preferred for making modifications.

1.14. “You” (or “Your”)

      means an individual or a legal entity exercising rights under this
      License. For legal entities, “You” includes any entity that controls, is
      controlled by, or is under common control with You. For purposes of this
      definition, “control” means (a) the power, direct or indirect, to cause
      the direction or management of such entity, whether by contract or
      otherwise, or (b) ownership of more than fifty percent (50%) of the
      outstanding shares or beneficial ownership of such entity.


2. License Grants and Conditions

2.1. Grants

     Each Contributor hereby grants You a world-wide, royalty-free,
     non-exclusive license:

     a. under intellectual property rights (other than patent or trademark)
        Licensable by such Contributor to use, reproduce, make available,
```

(continues on next page)

(continued from previous page)

```
        modify, display, perform, distribute, and otherwise exploit its
        Contributions, either on an unmodified basis, with Modifications, or as
        part of a Larger Work; and

     b. under Patent Claims of such Contributor to make, use, sell, offer for
        sale, have made, import, and otherwise transfer either its Contributions
        or its Contributor Version.

2.2. Effective Date

     The licenses granted in Section 2.1 with respect to any Contribution become
     effective for each Contribution on the date the Contributor first distributes
     such Contribution.

2.3. Limitations on Grant Scope

     The licenses granted in this Section 2 are the only rights granted under this
     License. No additional rights or licenses will be implied from the distribution
     or licensing of Covered Software under this License. Notwithstanding Section
     2.1(b) above, no patent license is granted by a Contributor:

     a. for any code that a Contributor has removed from Covered Software; or

     b. for infringements caused by: (i) Your and any other third party's
        modifications of Covered Software, or (ii) the combination of its
        Contributions with other software (except as part of its Contributor
        Version); or

     c. under Patent Claims infringed by Covered Software in the absence of its
        Contributions.

     This License does not grant any rights in the trademarks, service marks, or
     logos of any Contributor (except as may be necessary to comply with the
     notice requirements in Section 3.4).

2.4. Subsequent Licenses

     No Contributor makes additional grants as a result of Your choice to
     distribute the Covered Software under a subsequent version of this License
     (see Section 10.2) or under the terms of a Secondary License (if permitted
     under the terms of Section 3.3).

2.5. Representation

     Each Contributor represents that the Contributor believes its Contributions
     are its original creation(s) or it has sufficient rights to grant the
     rights to its Contributions conveyed by this License.

2.6. Fair Use

     This License is not intended to limit any rights You have under applicable
     copyright doctrines of fair use, fair dealing, or other equivalents.
```

(continues on next page)

(continued from previous page)

```
2.7. Conditions

     Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in
     Section 2.1.


3. Responsibilities

3.1. Distribution of Source Form

     All distribution of Covered Software in Source Code Form, including any
     Modifications that You create or to which You contribute, must be under the
     terms of this License. You must inform recipients that the Source Code Form
     of the Covered Software is governed by the terms of this License, and how
     they can obtain a copy of this License. You may not attempt to alter or
     restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

     If You distribute Covered Software in Executable Form then:

     a. such Covered Software must also be made available in Source Code Form,
        as described in Section 3.1, and You must inform recipients of the
        Executable Form how they can obtain a copy of such Source Code Form by
        reasonable means in a timely manner, at a charge no more than the cost
        of distribution to the recipient; and

     b. You may distribute such Executable Form under the terms of this License,
        or sublicense it under different terms, provided that the license for
        the Executable Form does not attempt to limit or alter the recipients'
        rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

     You may create and distribute a Larger Work under terms of Your choice,
     provided that You also comply with the requirements of this License for the
     Covered Software. If the Larger Work is a combination of Covered Software
     with a work governed by one or more Secondary Licenses, and the Covered
     Software is not Incompatible With Secondary Licenses, this License permits
     You to additionally distribute such Covered Software under the terms of
     such Secondary License(s), so that the recipient of the Larger Work may, at
     their option, further distribute the Covered Software under the terms of
     either this License or such Secondary License(s).

3.4. Notices

     You may not remove or alter the substance of any license notices (including
     copyright notices, patent notices, disclaimers of warranty, or limitations
     of liability) contained within the Source Code Form of the Covered
     Software, except that You may alter any license notices to the extent
     required to remedy known factual inaccuracies.
```

(continues on next page)

(continued from previous page)

```
3.5. Application of Additional Terms

     You may choose to offer, and to charge a fee for, warranty, support,
     indemnity or liability obligations to one or more recipients of Covered
     Software. However, You may do so only on Your own behalf, and not on behalf
     of any Contributor. You must make it absolutely clear that any such
     warranty, support, indemnity, or liability obligation is offered by You
     alone, and You hereby agree to indemnify every Contributor for any
     liability incurred by such Contributor as a result of warranty, support,
     indemnity or liability terms You offer. You may include additional
     disclaimers of warranty and limitations of liability specific to any
     jurisdiction.

4. Inability to Comply Due to Statute or Regulation

   If it is impossible for You to comply with any of the terms of this License
   with respect to some or all of the Covered Software due to statute, judicial
   order, or regulation then You must: (a) comply with the terms of this License
   to the maximum extent possible; and (b) describe the limitations and the code
   they affect. Such description must be placed in a text file included with all
   distributions of the Covered Software under this License. Except to the
   extent prohibited by statute or regulation, such description must be
   sufficiently detailed for a recipient of ordinary skill to be able to
   understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You
     fail to comply with any of its terms. However, if You become compliant,
     then the rights granted under this License from a particular Contributor
     are reinstated (a) provisionally, unless and until such Contributor
     explicitly and finally terminates Your grants, and (b) on an ongoing basis,
     if such Contributor fails to notify You of the non-compliance by some
     reasonable means prior to 60 days after You have come back into compliance.
     Moreover, Your grants from a particular Contributor are reinstated on an
     ongoing basis if such Contributor notifies You of the non-compliance by
     some reasonable means, this is the first time You have received notice of
     non-compliance with this License from such Contributor, and You become
     compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
     infringement claim (excluding declaratory judgment actions, counter-claims,
     and cross-claims) alleging that a Contributor Version directly or
     indirectly infringes any patent, then the rights granted to You by any and
     all Contributors for the Covered Software under Section 2.1 of this License
     shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user
     license agreements (excluding distributors and resellers) which have been
     validly granted by You or Your distributors under this License prior to
     termination shall survive termination.
```

(continues on next page)

(continued from previous page)

```
6. Disclaimer of Warranty

   Covered Software is provided under this License on an “as is” basis, without
   warranty of any kind, either expressed, implied, or statutory, including,
   without limitation, warranties that the Covered Software is free of defects,
   merchantable, fit for a particular purpose or non-infringing. The entire
   risk as to the quality and performance of the Covered Software is with You.
   Should any Covered Software prove defective in any respect, You (not any
   Contributor) assume the cost of any necessary servicing, repair, or
   correction. This disclaimer of warranty constitutes an essential part of this
   License. No use of  any Covered Software is authorized under this License
   except under this disclaimer.

7. Limitation of Liability

   Under no circumstances and under no legal theory, whether tort (including
   negligence), contract, or otherwise, shall any Contributor, or anyone who
   distributes Covered Software as permitted above, be liable to You for any
   direct, indirect, special, incidental, or consequential damages of any
   character including, without limitation, damages for lost profits, loss of
   goodwill, work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses, even if such party shall have been
   informed of the possibility of such damages. This limitation of liability
   shall not apply to liability for death or personal injury resulting from such
   party’s negligence to the extent applicable law prohibits such limitation.
   Some jurisdictions do not allow the exclusion or limitation of incidental or
   consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

   Any litigation relating to this License may be brought only in the courts of
   a jurisdiction where the defendant maintains its principal place of business
   and such litigation shall be governed by laws of that jurisdiction, without
   reference to its conflict-of-law provisions. Nothing in this Section shall
   prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

   This License represents the complete agreement concerning the subject matter
   hereof. If any provision of this License is held to be unenforceable, such
   provision shall be reformed only to the extent necessary to make it
   enforceable. Any law or regulation which provides that the language of a
   contract shall be construed against the drafter shall not be used to construe
   this License against a Contributor.


10. Versions of the License

10.1. New Versions

      Mozilla Foundation is the license steward. Except as provided in Section
```

(continues on next page)

(continued from previous page)

```
      10.3, no one other than the license steward has the right to modify or
      publish new versions of this License. Each version will be given a
      distinguishing version number.

10.2. Effect of New Versions

      You may distribute the Covered Software under the terms of the version of
      the License under which You originally received the Covered Software, or
      under the terms of any subsequent version published by the license
      steward.

10.3. Modified Versions

      If you create software not governed by this License, and you want to
      create a new license for such software, you may create and use a modified
      version of this License if you rename the license and remove any
      references to the name of the license steward (except to note that such
      modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses
      If You choose to distribute Source Code Form that is Incompatible With
      Secondary Licenses under the terms of this version of the License, the
      notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

      This Source Code Form is subject to the
      terms of the Mozilla Public License, v.
      2.0. If a copy of the MPL was not
      distributed with this file, You can
      obtain one at
      http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then
You may include the notice in a location (such as a LICENSE file in a relevant
directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - “Incompatible With Secondary Licenses” Notice

      This Source Code Form is “Incompatible
      With Secondary Licenses”, as defined by
      the Mozilla Public License, v. 2.0.
```

github.com/josharian/intern

License: MIT

```
MIT License

Copyright (c) 2019 Josh Bleecher Snyder

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/json-iterator/go

License: MIT

```
MIT License

Copyright (c) 2016 json-iterator

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/klauspost/compress

License: Apache-2.0

```
Copyright (c) 2012 The Go Authors. All rights reserved.
Copyright (c) 2019 Klaus Post. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------

Files: gzhttp/*

                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
```

(continues on next page)

(continued from previous page)

```
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

      "You" (or "Your") shall mean an individual or Legal Entity
      exercising permissions granted by this License.

      "Source" form shall mean the preferred form for making modifications,
      including but not limited to software source code, documentation
      source, and configuration files.

      "Object" form shall mean any form resulting from mechanical
      transformation or translation of a Source form, including but
      not limited to compiled object code, generated documentation,
      and conversions to other media types.

      "Work" shall mean the work of authorship, whether in Source or
      Object form, made available under the License, as indicated by a
      copyright notice that is included in or attached to the work
      (an example is provided in the Appendix below).

      "Derivative Works" shall mean any work, whether in Source or Object
      form, that is based on (or derived from) the Work and for which the
      editorial revisions, annotations, elaborations, or other modifications
      represent, as a whole, an original work of authorship. For the purposes
      of this License, Derivative Works shall not include works that remain
      separable from, or merely link (or bind by name) to the interfaces of,
      the Work and Derivative Works thereof.

      "Contribution" shall mean any work of authorship, including
      the original version of the Work and any modifications or additions
      to that Work or Derivative Works thereof, that is intentionally
      submitted to Licensor for inclusion in the Work by the copyright owner
      or by an individual or Legal Entity authorized to submit on behalf of
      the copyright owner. For the purposes of this definition, "submitted"
      means any form of electronic, verbal, or written communication sent
      to the Licensor or its representatives, including but not limited to
      communication on electronic mailing lists, source code control systems,
      and issue tracking systems that are managed by, or on behalf of, the
      Licensor for the purpose of discussing and improving the Work, but
      excluding communication that is conspicuously marked or otherwise
      designated in writing by the copyright owner as "Not a Contribution."

      "Contributor" shall mean Licensor and any individual or Legal Entity
      on behalf of whom a Contribution has been received by Licensor and
      subsequently incorporated within the Work.

   2. Grant of Copyright License. Subject to the terms and conditions of
      this License, each Contributor hereby grants to You a perpetual,
      worldwide, non-exclusive, no-charge, royalty-free, irrevocable
```

(continues on next page)

(continued from previous page)

```
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
```

(continues on next page)

(continued from previous page)

```
        as modifying the License.

    You may add Your own copyright statement to Your modifications and
    may provide additional or different license terms and conditions
    for use, reproduction, or distribution of Your modifications, or
    for any such Derivative Works as a whole, provided Your use,
    reproduction, and distribution of the Work otherwise complies with
    the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
    any Contribution intentionally submitted for inclusion in the Work
    by You to the Licensor shall be under the terms and conditions of
    this License, without any additional terms or conditions.
    Notwithstanding the above, nothing herein shall supersede or modify
    the terms of any separate license agreement you may have executed
    with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
    names, trademarks, service marks, or product names of the Licensor,
    except as required for reasonable and customary use in describing the
    origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
    agreed to in writing, Licensor provides the Work (and each
    Contributor provides its Contributions) on an "AS IS" BASIS,
    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
    implied, including, without limitation, any warranties or conditions
    of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
    PARTICULAR PURPOSE. You are solely responsible for determining the
    appropriateness of using or redistributing the Work and assume any
    risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
    whether in tort (including negligence), contract, or otherwise,
    unless required by applicable law (such as deliberate and grossly
    negligent acts) or agreed to in writing, shall any Contributor be
    liable to You for damages, including any direct, indirect, special,
    incidental, or consequential damages of any character arising as a
    result of this License or out of the use or inability to use the
    Work (including but not limited to damages for loss of goodwill,
    work stoppage, computer failure or malfunction, or any and all
    other commercial damages or losses), even if such Contributor
    has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
    the Work or Derivative Works thereof, You may choose to offer,
    and charge a fee for, acceptance of support, warranty, indemnity,
    or other liability obligations and/or rights consistent with this
    License. However, in accepting such obligations, You may act only
    on Your own behalf and on Your sole responsibility, not on behalf
    of any other Contributor, and only if You agree to indemnify,
    defend, and hold each Contributor harmless for any liability
```

(continues on next page)

(continued from previous page)

```
      incurred by, or claims asserted against, such Contributor by reason
      of your accepting any such warranty or additional liability.

   END OF TERMS AND CONDITIONS

   APPENDIX: How to apply the Apache License to your work.

      To apply the Apache License to your work, attach the following
      boilerplate notice, with the fields enclosed by brackets "[]"
      replaced with your own identifying information. (Don't include
      the brackets!)  The text should be enclosed in the appropriate
      comment syntax for the file format. We also recommend that a
      file or class name and description of purpose be included on the
      same "printed page" as the copyright notice for easier
      identification within third-party archives.

   Copyright 2016-2017 The New York Times Company

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.

------------------

Files: s2/cmd/internal/readahead/*

The MIT License (MIT)

Copyright (c) 2015 Klaus Post

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
```

(continues on next page)

(continued from previous page)

```
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

----------------------
Files: snappy/*
Files: internal/snapref/*

Copyright (c) 2011 The Snappy-Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-----------------

Files: s2/cmd/internal/filepathx/*

Copyright 2016 The filepathx Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this␣
↪software and associated documentation files (the "Software"), to deal in the Software␣
↪without restriction, including without limitation the rights to use, copy, modify,␣
↪merge, publish, distribute, sublicense, and/or sell copies of the Software, and to␣
↪permit persons to whom the Software is furnished to do so, subject to the following␣
↪conditions:

The above copyright notice and this permission notice shall be included in all copies or␣
↪substantial portions of the Software.
```

(continues on next page)

(continued from previous page)

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,␣
↪INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A␣
↪PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT␣
↪HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF␣
↪CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE␣
↪OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/klauspost/compress/internal/snapref

License: BSD-3-Clause

```
Copyright (c) 2011 The Snappy-Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/klauspost/compress/zstd/internal/xxhash

License: MIT

```
Copyright (c) 2016 Caleb Spare

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
```

(continues on next page)

(continued from previous page)

```
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/klauspost/pgzip

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014 Klaus Post

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/letsencrypt/boulder

License: MPL-2.0

```
Copyright 2016 ISRG.  All rights reserved.

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to
    the creation of, or owns Covered Software.

1.2. "Contributor Version"
    means the combination of the Contributions of others (if any) used
    by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.

1.4. "Covered Software"
    means Source Code Form to which the initial Contributor has attached
    the notice in Exhibit A, the Executable Form of such Source Code
    Form, and Modifications of such Source Code Form, in each case
    including portions thereof.

1.5. "Incompatible With Secondary Licenses"
    means

    (a) that the initial Contributor has attached the notice described
        in Exhibit B to the Covered Software; or

    (b) that the Covered Software was made available under the terms of
        version 1.1 or earlier of the License, but not also under the
        terms of a Secondary License.

1.6. "Executable Form"
    means any form of the work other than Source Code Form.

1.7. "Larger Work"
    means a work that combines Covered Software with other material, in
    a separate file or files, that is not Covered Software.

1.8. "License"
    means this document.

1.9. "Licensable"
    means having the right to grant, to the maximum extent possible,
    whether at the time of the initial grant or subsequently, any and
    all of the rights conveyed by this License.
```

(continues on next page)

(continued from previous page)

```
1.10. "Modifications"
    means any of the following:

    (a) any file in Source Code Form that results from an addition to,
        deletion from, or modification of the contents of Covered
        Software; or

    (b) any new file in Source Code Form that contains any Covered
        Software.

1.11. "Patent Claims" of a Contributor
    means any patent claim(s), including without limitation, method,
    process, and apparatus claims, in any patent Licensable by such
    Contributor that would be infringed, but for the grant of the
    License, by the making, using, selling, offering for sale, having
    made, import, or transfer of either its Contributions or its
    Contributor Version.

1.12. "Secondary License"
    means either the GNU General Public License, Version 2.0, the GNU
    Lesser General Public License, Version 2.1, the GNU Affero General
    Public License, Version 3.0, or any later versions of those
    licenses.

1.13. "Source Code Form"
    means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
    means an individual or a legal entity exercising rights under this
    License. For legal entities, "You" includes any entity that
    controls, is controlled by, or is under common control with You. For
    purposes of this definition, "control" means (a) the power, direct
    or indirect, to cause the direction or management of such entity,
    whether by contract or otherwise, or (b) ownership of more than
    fifty percent (50%) of the outstanding shares or beneficial
    ownership of such entity.

2. License Grants and Conditions
--------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
    Licensable by such Contributor to use, reproduce, make available,
    modify, display, perform, distribute, and otherwise exploit its
    Contributions, either on an unmodified basis, with Modifications, or
    as part of a Larger Work; and
```

(continues on next page)

(continued from previous page)

```
(b) under Patent Claims of such Contributor to make, use, sell, offer
    for sale, have made, import, and otherwise transfer either its
    Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License.
Notwithstanding Section 2.1(b) above, no patent license is granted by a
Contributor:

(a) for any code that a Contributor has removed from Covered Software;
    or

(b) for infringements caused by: (i) Your and any other third party's
    modifications of Covered Software, or (ii) the combination of its
    Contributions with other software (except as part of its Contributor
    Version); or

(c) under Patent Claims infringed by Covered Software in the absence of
    its Contributions.

This License does not grant any rights in the trademarks, service marks,
or logos of any Contributor (except as may be necessary to comply with
the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to
distribute the Covered Software under a subsequent version of this
License (see Section 10.2) or under the terms of a Secondary License (if
permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its
Contributions are its original creation(s) or it has sufficient rights
to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under
applicable copyright doctrines of fair use, fair dealing, or other
equivalents.
```

(continues on next page)

(continued from previous page)

```
2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted
in Section 2.1.

3. Responsibilities
-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any
Modifications that You create or to which You contribute, must be under
the terms of this License. You must inform recipients that the Source
Code Form of the Covered Software is governed by the terms of this
License, and how they can obtain a copy of this License. You may not
attempt to alter or restrict the recipients' rights in the Source Code
Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code
    Form, as described in Section 3.1, and You must inform recipients of
    the Executable Form how they can obtain a copy of such Source Code
    Form by reasonable means in a timely manner, at a charge no more
    than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this
    License, or sublicense it under different terms, provided that the
    license for the Executable Form does not attempt to limit or alter
    the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice,
provided that You also comply with the requirements of this License for
the Covered Software. If the Larger Work is a combination of Covered
Software with a work governed by one or more Secondary Licenses, and the
Covered Software is not Incompatible With Secondary Licenses, this
License permits You to additionally distribute such Covered Software
under the terms of such Secondary License(s), so that the recipient of
the Larger Work may, at their option, further distribute the Covered
Software under the terms of either this License or such Secondary
License(s).

3.4. Notices

You may not remove or alter the substance of any license notices
(including copyright notices, patent notices, disclaimers of warranty,
or limitations of liability) contained within the Source Code Form of
the Covered Software, except that You may alter any license notices to
```

(continues on next page)

(continued from previous page)

```
the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support,
indemnity or liability obligations to one or more recipients of Covered
Software. However, You may do so only on Your own behalf, and not on
behalf of any Contributor. You must make it absolutely clear that any
such warranty, support, indemnity, or liability obligation is offered by
You alone, and You hereby agree to indemnify every Contributor for any
liability incurred by such Contributor as a result of warranty, support,
indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation
---------------------------------------------------

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Software due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description must
be placed in a text file included with all distributions of the Covered
Software under this License. Except to the extent prohibited by statute
or regulation, such description must be sufficiently detailed for a
recipient of ordinary skill to be able to understand it.

5. Termination
--------------

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.
```

(continues on next page)

(continued from previous page)

```
5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  ------------------------                                            *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
```

(continues on next page)

(continued from previous page)

```
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous
----------------

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License
---------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
of the License under which You originally received the Covered Software,
or under the terms of any subsequent version published by the license
steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a
modified version of this License if you rename the license and remove
any references to the name of the license steward (except to note that
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary
Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

  This Source Code Form is subject to the terms of the Mozilla Public
  License, v. 2.0. If a copy of the MPL was not distributed with this
```

(continues on next page)

(continued from previous page)

```
  file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

  This Source Code Form is "Incompatible With Secondary Licenses", as
  defined by the Mozilla Public License, v. 2.0.
```

github.com/mailru/easyjson

License: MIT

```
Copyright (c) 2016 Mail.Ru Group

Permission is hereby granted, free of charge, to any person obtaining a copy of this␣
↪software and associated documentation files (the "Software"), to deal in the Software␣
↪without restriction, including without limitation the rights to use, copy, modify,␣
↪merge, publish, distribute, sublicense, and/or sell copies of the Software, and to␣
↪permit persons to whom the Software is furnished to do so, subject to the following␣
↪conditions:

The above copyright notice and this permission notice shall be included in all copies or␣
↪substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,␣
↪INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A␣
↪PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT␣
↪HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF␣
↪CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE␣
↪OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/mattn/go-colorable

License: MIT

```
The MIT License (MIT)

Copyright (c) 2016 Yasuhiro Matsumoto

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
```

(continues on next page)

(continued from previous page)

```
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/mattn/go-isatty

License: MIT

```
Copyright (c) Yasuhiro MATSUMOTO <mattn.jp@gmail.com>

MIT License (Expat)

Permission is hereby granted, free of charge, to any person obtaining a copy of this␣
↪software and associated documentation files (the "Software"), to deal in the Software␣
↪without restriction, including without limitation the rights to use, copy, modify,␣
↪merge, publish, distribute, sublicense, and/or sell copies of the Software, and to␣
↪permit persons to whom the Software is furnished to do so, subject to the following␣
↪conditions:

The above copyright notice and this permission notice shall be included in all copies or␣
↪substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,␣
↪INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A␣
↪PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT␣
↪HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF␣
↪CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE␣
↪OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

github.com/mattn/go-runewidth

License: MIT

```
The MIT License (MIT)

Copyright (c) 2016 Yasuhiro Matsumoto

Permission is hereby granted, free of charge, to any person obtaining a copy
```

(continues on next page)

(continued from previous page)

```
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/mattn/go-sqlite3

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014 Yasuhiro Matsumoto

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/matttproud/golang_protobuf_extensions/pbutil

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/miekg/pkcs11

License: BSD-3-Clause

```
Copyright (c) 2013 Miek Gieben. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Miek Gieben nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/mitchellh/go-homedir

License: MIT

```
The MIT License (MIT)

Copyright (c) 2013 Mitchell Hashimoto

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
```

(continues on next page)

(continued from previous page)

```
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/mitchellh/mapstructure

License: MIT

```
The MIT License (MIT)

Copyright (c) 2013 Mitchell Hashimoto

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/moby/patternmatcher

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.
```

(continues on next page)

(continued from previous page)

```
"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
```

(continues on next page)

(continued from previous page)

```
2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
```

(continues on next page)

(continued from previous page)

```
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
```

(continues on next page)

(continued from previous page)

```
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2018 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/moby/sys/mountinfo

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.
```

(continues on next page)

(continued from previous page)

```
   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
```

(continues on next page)

(continued from previous page)

```
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.
```

(continues on next page)

(continued from previous page)

```
5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
```

(continues on next page)

(continued from previous page)

```
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/moby/sys/sequential

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
```

(continues on next page)

(continued from previous page)

```
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
```

(continues on next page)

(continued from previous page)

```
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
```

(continues on next page)

(continued from previous page)

```
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
```

(continues on next page)

(continued from previous page)

```
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/moby/term

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
```

(continues on next page)

(continued from previous page)

```
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
```

(continues on next page)

(continued from previous page)

```
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
```

(continues on next page)

(continued from previous page)

```
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2013-2018 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
```

(continues on next page)

(continued from previous page)

```
    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/modern-go/concurrent

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).
```

(continues on next page)

(continued from previous page)

```
   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
```

(continues on next page)

(continued from previous page)

```
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
```

(continues on next page)

(continued from previous page)

```
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/modern-go/reflect2

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
```

(continues on next page)

(continued from previous page)

```
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
```

(continues on next page)

(continued from previous page)

```
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/oklog/ulid

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
```

(continues on next page)

(continued from previous page)

```
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
```

(continues on next page)

(continued from previous page)

```
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
```

(continues on next page)

(continued from previous page)

```
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/go-digest

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
```

(continues on next page)

(continued from previous page)

```
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
```

(continues on next page)

(continued from previous page)

```
      Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
```

(continues on next page)

(continued from previous page)

```
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2019, 2020 OCI Contributors
Copyright 2016 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/image-spec/specs-go

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2016 The Linux Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/runc/libcontainer

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
```

(continues on next page)

(continued from previous page)

```
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."
```

(continues on next page)

(continued from previous page)

```
   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
```

(continues on next page)

(continued from previous page)

```
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
```

(continues on next page)

(continued from previous page)

```
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2014 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/runtime-spec/specs-go

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.
```

(continues on next page)

(continued from previous page)

```
   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
```

(continues on next page)

(continued from previous page)

```
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
```

(continues on next page)

(continued from previous page)

```
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

(continues on next page)

(continued from previous page)

```
Copyright 2015 The Linux Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/runtime-tools

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.
```

(continues on next page)

(continued from previous page)

```
   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
```

(continues on next page)

(continued from previous page)

```
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.
```

(continues on next page)

(continued from previous page)

```
6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2015 The Linux Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
```

(continues on next page)

(continued from previous page)

```
limitations under the License.
```

github.com/opencontainers/selinux

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
```

(continues on next page)

(continued from previous page)

```
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
```

(continues on next page)

(continued from previous page)

```
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
```

(continues on next page)

(continued from previous page)

```
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/umoci

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/opencontainers/umoci/third_party/shared

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/pelletier/go-toml/v2

License: MIT

```
The MIT License (MIT)

go-toml v2
Copyright (c) 2021 - 2023 Thomas Pelletier

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/pkg/errors

License: BSD-2-Clause

```
Copyright (c) 2015, Dave Cheney <dave@cheney.net>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
```

(continues on next page)

(continued from previous page)

```
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/proglottis/gpgme

License: BSD-3-Clause

```
Copyright (c) 2015, James Fargher <proglottis@gmail.com>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are␣
↪permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of␣
↪conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list␣
↪of conditions and the following disclaimer in the documentation and/or other materials␣
↪provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be␣
↪used to endorse or promote products derived from this software without specific prior␣
↪written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY␣
↪EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES␣
↪OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT␣
↪SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,␣
↪INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED␣
↪TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR␣
↪BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN␣
↪CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY␣
↪WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/prometheus/client_golang/prometheus

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.
```

(continues on next page)

(continued from previous page)

```
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
```

(continues on next page)

(continued from previous page)

```
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
```

(continues on next page)

(continued from previous page)

```
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
```

(continues on next page)

(continued from previous page)

```
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/prometheus/client_model/go

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
```

(continues on next page)

(continued from previous page)

```
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.
```

(continues on next page)

(continued from previous page)

```
2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
```

(continues on next page)

(continued from previous page)

```
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
```

(continues on next page)

(continued from previous page)

```
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/prometheus/common

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.
```

(continues on next page)

(continued from previous page)

```
"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.
```

(continues on next page)

(continued from previous page)

```
2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
```

(continues on next page)

(continued from previous page)

```
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
```

(continues on next page)

(continued from previous page)

```
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/prometheus/common/internal/bitbucket.org/ww/goautoneg

License: BSD-3-Clause

```
PACKAGE

package goautoneg
import "bitbucket.org/ww/goautoneg"

HTTP Content-Type Autonegotiation.

The functions in this package implement the behaviour specified in
http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html

Copyright (c) 2011, Open Knowledge Foundation Ltd.
All rights reserved.

Redistribution and use in source and binary forms, with or without
```

(continues on next page)

(continued from previous page)

```
modification, are permitted provided that the following conditions are
met:

    Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.

    Redistributions in binary form must reproduce the above copyright
    notice, this list of conditions and the following disclaimer in
    the documentation and/or other materials provided with the
    distribution.

    Neither the name of the Open Knowledge Foundation Ltd. nor the
    names of its contributors may be used to endorse or promote
    products derived from this software without specific prior written
    permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.


FUNCTIONS

func Negotiate(header string, alternatives []string) (content_type string)
Negotiate the most appropriate content_type given the accept header
and a list of alternatives.

func ParseAccept(header string) (accept []Accept)
Parse an Accept Header string returning a sorted list
of clauses


TYPES

type Accept struct {
    Type, SubType string
    Q             float32
    Params        map[string]string
}
Structure to represent a clause in an HTTP Accept Header


SUBDIRECTORIES
```

(continues on next page)

(continued from previous page)

```
.hg
```

github.com/prometheus/procfs

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
```

(continues on next page)

(continued from previous page)

```
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
```

(continues on next page)

(continued from previous page)

```
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
```

(continues on next page)

(continued from previous page)

```
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/rivo/uniseg

License: MIT

```
MIT License

Copyright (c) 2019 Oliver Kuederle

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/rootless-containers/proto/go-proto

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.
```

(continues on next page)

(continued from previous page)

```
   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
```

(continues on next page)

(continued from previous page)

```
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
```

(continues on next page)

(continued from previous page)

```
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
```

(continues on next page)

(continued from previous page)

```
   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/russross/blackfriday/v2

License: BSD-2-Clause

```
Blackfriday is distributed under the Simplified BSD License:

> Copyright © 2011 Russ Ross
> All rights reserved.
>
> Redistribution and use in source and binary forms, with or without
> modification, are permitted provided that the following conditions
> are met:
>
> 1.  Redistributions of source code must retain the above copyright
>     notice, this list of conditions and the following disclaimer.
>
> 2.  Redistributions in binary form must reproduce the above
>     copyright notice, this list of conditions and the following
>     disclaimer in the documentation and/or other materials provided with
>     the distribution.
>
> THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
> "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
> LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
> FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
> COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
> INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
```

(continues on next page)

(continued from previous page)

```
> BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
> LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
> CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
> LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
> ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
> POSSIBILITY OF SUCH DAMAGE.
```

github.com/safchain/ethtool

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
```

(continues on next page)

(continued from previous page)

```
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
```

(continues on next page)

(continued from previous page)

```
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/samber/lo

License: MIT

```
MIT License

Copyright (c) 2022 Samuel Berthe

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/secure-systems-lab/go-securesystemslib

License: MIT

```
The MIT License (MIT)

Copyright (c) 2021 NYU Secure Systems Lab

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

(continues on next page)

(continued from previous page)

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/shopspring/decimal

License: MIT

```
The MIT License (MIT)

Copyright (c) 2015 Spring, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

- Based on https://github.com/oguzbilgic/fpd, which has the following license:
"""
The MIT License (MIT)

Copyright (c) 2013 Oguz Bilgic

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

(continues on next page)

(continued from previous page)

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
"""
```

github.com/sigstore/fulcio/pkg/certificate

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).
```

(continues on next page)

(continued from previous page)

```
   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
```

(continues on next page)

(continued from previous page)

```
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
```

(continues on next page)

(continued from previous page)

```
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/sigstore/rekor/pkg/generated/models

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
```

(continues on next page)

(continued from previous page)

```
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
```

(continues on next page)

(continued from previous page)

```
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/sigstore/sigstore/pkg

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
```

(continues on next page)

(continued from previous page)

```
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
```

(continues on next page)

(continued from previous page)

```
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
```

(continues on next page)

(continued from previous page)

```
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/sirupsen/logrus

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014 Simon Eskildsen

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

github.com/spf13/cobra

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.
```

(continues on next page)

(continued from previous page)

```
   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
```

(continues on next page)

(continued from previous page)

```
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
```

(continues on next page)

(continued from previous page)

```
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
```

(continues on next page)

(continued from previous page)

```
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.
```

github.com/spf13/pflag

License: BSD-3-Clause

```
Copyright (c) 2012 Alex Ogier. All rights reserved.
Copyright (c) 2012 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/stefanberger/go-pkcs11uri

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
```

(continues on next page)

(continued from previous page)

```
   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
```

(continues on next page)

(continued from previous page)

```
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
```

(continues on next page)

(continued from previous page)

```
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.
```

(continues on next page)

(continued from previous page)

```
9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

github.com/sylabs/json-resp

License: BSD-3-Clause

```
# LICENSE

Copyright (c) 2018-2022, Sylabs Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

1. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

1. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from this
   software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
```

github.com/sylabs/oci-tools/pkg

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
      attribution notices from the Source form of the Work,
      excluding those notices that do not pertain to any part of
      the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
      distribution, then any Derivative Works that You distribute must
      include a readable copy of the attribution notices contained
      within such NOTICE file, excluding those notices that do not
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/sylabs/scs-build-client/client

License: BSD-3-Clause

```
# LICENSE

Copyright (c) 2018-2022, Sylabs Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are␣
↪permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of␣
↪conditions and the following disclaimer.

1. Redistributions in binary form must reproduce the above copyright notice, this list␣
↪of conditions and the following disclaimer in the documentation and/or other materials␣
↪provided with the distribution.

1. Neither the name of the copyright holder nor the names of its contributors may be␣
↪used to endorse or promote products derived from this software without specific prior␣
↪written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY␣
↪EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES␣
↪OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT␣
↪SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,␣
↪INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED␣
↪TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR␣
↪BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN␣
↪CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY␣
↪WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/sylabs/scs-key-client/client

License: BSD-3-Clause

```
# LICENSE

Copyright (c) 2019-2022, Sylabs Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are␣
↪permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of␣
↪conditions and the following disclaimer.

1. Redistributions in binary form must reproduce the above copyright notice, this list␣
↪of conditions and the following disclaimer in the documentation and/or other materials␣
↪provided with the distribution.

1. Neither the name of the copyright holder nor the names of its contributors may be␣
↪used to endorse or promote products derived from this software without specific prior␣
```

(continues on next page)

(continued from previous page)

```
↪written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY␣
↪EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES␣
↪OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT␣
↪SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,␣
↪INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED␣
↪TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR␣
↪BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN␣
↪CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY␣
↪WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/sylabs/scs-library-client/client

License: BSD-3-Clause

```
# LICENSE

Copyright (c) 2018-2022, Sylabs Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are␣
↪permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of␣
↪conditions and the following disclaimer.

1. Redistributions in binary form must reproduce the above copyright notice, this list␣
↪of conditions and the following disclaimer in the documentation and/or other materials␣
↪provided with the distribution.

1. Neither the name of the copyright holder nor the names of its contributors may be␣
↪used to endorse or promote products derived from this software without specific prior␣
↪written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY␣
↪EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES␣
↪OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT␣
↪SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,␣
↪INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED␣
↪TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR␣
↪BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN␣
↪CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY␣
↪WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/sylabs/sif/v2

License: BSD-3-Clause

```
# LICENSE

Copyright (c) 2018-2023, Sylabs Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from this
   software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
```

github.com/syndtr/gocapability/capability

License: BSD-2-Clause

```
Copyright 2013 Suryandaru Triandana <syndtr@gmail.com>
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
```

(continues on next page)

(continued from previous page)

```
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/theupdateframework/go-tuf/encrypted

License: BSD-3-Clause

```
Copyright (c) 2014-2020 Prime Directive, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Prime Directive, Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/titanous/rocacheck

License: MIT

```
MIT License

Copyright (c) 2017, Jonathan Rudenberg
Copyright (c) 2017, CRoCS, EnigmaBridge Ltd.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/ulikunitz/xz

License: BSD-3-Clause

```
Copyright (c) 2014-2022  Ulrich Kunitz
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* My name, Ulrich Kunitz, may not be used to endorse or promote products
  derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
```

(continues on next page)

(continued from previous page)

```
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/urfave/cli

License: MIT

```
MIT License

Copyright (c) 2023 Jeremy Saenz & Contributors

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

github.com/vbatts/go-mtree

License: BSD-3-Clause

```
Copyright (c) 2016 Vincent Batts, Raleigh, NC, USA

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.
```

(continues on next page)

(continued from previous page)

```
3. Neither the name of the copyright holder nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/vbatts/go-mtree/pkg/govis

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
```

(continues on next page)

(continued from previous page)

```
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
```

(continues on next page)

(continued from previous page)

```
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
```

(continues on next page)

(continued from previous page)

```
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.
```

(continues on next page)

(continued from previous page)

```
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/vbatts/tar-split

License: BSD-3-Clause

```
Copyright (c) 2015 Vincent Batts, Raleigh, NC, USA

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

github.com/vbauerster/mpb/v8

License: Unlicense

```
This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or
distribute this software, either in source code form or as a compiled
binary, for any purpose, commercial or non-commercial, and by any
means.

In jurisdictions that recognize copyright laws, the author or authors
of this software dedicate any and all copyright interest in the
software to the public domain. We make this dedication for the benefit
of the public at large and to the detriment of our heirs and
successors. We intend this dedication to be an overt act of
relinquishment in perpetuity of all present and future rights to this
software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>
```

github.com/vishvananda/netlink

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
```

(continues on next page)

(continued from previous page)

```
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.
```

(continues on next page)

(continued from previous page)

```
3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
```

(continues on next page)

(continued from previous page)

```
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.
```

(continues on next page)

(continued from previous page)

```
END OF TERMS AND CONDITIONS

Copyright 2014 Vishvananda Ishaya.
Copyright 2014 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/vishvananda/netns

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
```

(continues on next page)

(continued from previous page)

```
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
```

(continues on next page)

(continued from previous page)

```
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
```

(continues on next page)

(continued from previous page)

```
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2014 Vishvananda Ishaya.
Copyright 2014 Docker, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

(continues on next page)

(continued from previous page)

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/xeipuuv/gojsonpointer

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
```

(continues on next page)

(continued from previous page)

```
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:
```

(continues on next page)

(continued from previous page)

```
   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
```

(continues on next page)

(continued from previous page)

```
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2015 xeipuuv

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
```

(continues on next page)

(continued from previous page)

```
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/xeipuuv/gojsonreference

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
```

(continues on next page)

(continued from previous page)

```
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
```

(continues on next page)

(continued from previous page)

```
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
```

(continues on next page)

(continued from previous page)

```
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2015 xeipuuv

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
```

(continues on next page)

(continued from previous page)

```
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

github.com/xeipuuv/gojsonschema

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
```

(continues on next page)

(continued from previous page)

```
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and
```

(continues on next page)

(continued from previous page)

```
   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
```

(continues on next page)

(continued from previous page)

```
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright 2015 xeipuuv

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

(continues on next page)

(continued from previous page)

```
See the License for the specific language governing permissions and
limitations under the License.
```

go.mongodb.org/mongo-driver

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
```

(continues on next page)

(continued from previous page)

```
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and
```

(continues on next page)

(continued from previous page)

```
   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
```

(continues on next page)

(continued from previous page)

```
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
```

(continues on next page)

(continued from previous page)

```
limitations under the License.
```

go.mozilla.org/pkcs7

License: MIT

```
The MIT License (MIT)

Copyright (c) 2015 Andrew Smith

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

golang.org/x/crypto

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
```

(continues on next page)

(continued from previous page)

```
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/exp

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/mod/semver

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/net

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.
```

(continues on next page)

(continued from previous page)

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/sync

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/sys

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/term

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.
```

(continues on next page)

(continued from previous page)

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

golang.org/x/text

License: BSD-3-Clause

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

google.golang.org/genproto/googleapis/rpc/status

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

google.golang.org/grpc

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
```

(continues on next page)

(continued from previous page)

```
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
```

(continues on next page)

(continued from previous page)

```
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
```

(continues on next page)

(continued from previous page)

```
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

google.golang.org/protobuf

License: BSD-3-Clause

```
Copyright (c) 2018 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

gopkg.in/go-jose/go-jose.v2

License: Apache-2.0

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
```

(continues on next page)

(continued from previous page)

```
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
```

(continues on next page)

(continued from previous page)

```
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
```

(continues on next page)

(continued from previous page)

```
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
```

(continues on next page)

(continued from previous page)

```
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

gopkg.in/go-jose/go-jose.v2/json

License: BSD-3-Clause

```
Copyright (c) 2012 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
```

(continues on next page)

(continued from previous page)

```
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

gopkg.in/yaml.v2

License: Apache-2.0

```
                              Apache License
                        Version 2.0, January 2004
                     http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   "License" shall mean the terms and conditions for use, reproduction,
   and distribution as defined by Sections 1 through 9 of this document.

   "Licensor" shall mean the copyright owner or entity authorized by
   the copyright owner that is granting the License.

   "Legal Entity" shall mean the union of the acting entity and all
   other entities that control, are controlled by, or are under common
   control with that entity. For the purposes of this definition,
   "control" means (i) the power, direct or indirect, to cause the
   direction or management of such entity, whether by contract or
   otherwise, or (ii) ownership of fifty percent (50%) or more of the
   outstanding shares, or (iii) beneficial ownership of such entity.

   "You" (or "Your") shall mean an individual or Legal Entity
   exercising permissions granted by this License.

   "Source" form shall mean the preferred form for making modifications,
   including but not limited to software source code, documentation
   source, and configuration files.

   "Object" form shall mean any form resulting from mechanical
   transformation or translation of a Source form, including but
   not limited to compiled object code, generated documentation,
   and conversions to other media types.

   "Work" shall mean the work of authorship, whether in Source or
   Object form, made available under the License, as indicated by a
```

(continues on next page)

(continued from previous page)

```
   copyright notice that is included in or attached to the work
   (an example is provided in the Appendix below).

   "Derivative Works" shall mean any work, whether in Source or Object
   form, that is based on (or derived from) the Work and for which the
   editorial revisions, annotations, elaborations, or other modifications
   represent, as a whole, an original work of authorship. For the purposes
   of this License, Derivative Works shall not include works that remain
   separable from, or merely link (or bind by name) to the interfaces of,
   the Work and Derivative Works thereof.

   "Contribution" shall mean any work of authorship, including
   the original version of the Work and any modifications or additions
   to that Work or Derivative Works thereof, that is intentionally
   submitted to Licensor for inclusion in the Work by the copyright owner
   or by an individual or Legal Entity authorized to submit on behalf of
   the copyright owner. For the purposes of this definition, "submitted"
   means any form of electronic, verbal, or written communication sent
   to the Licensor or its representatives, including but not limited to
   communication on electronic mailing lists, source code control systems,
   and issue tracking systems that are managed by, or on behalf of, the
   Licensor for the purpose of discussing and improving the Work, but
   excluding communication that is conspicuously marked or otherwise
   designated in writing by the copyright owner as "Not a Contribution."

   "Contributor" shall mean Licensor and any individual or Legal Entity
   on behalf of whom a Contribution has been received by Licensor and
   subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   (except as stated in this section) patent license to make, have made,
   use, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.
```

(continues on next page)

(continued from previous page)

```
4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works
       that You distribute, all copyright, patent, trademark, and
       attribution notices from the Source form of the Work,
       excluding those notices that do not pertain to any part of
       the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its
       distribution, then any Derivative Works that You distribute must
       include a readable copy of the attribution notices contained
       within such NOTICE file, excluding those notices that do not
       pertain to any part of the Derivative Works, in at least one
       of the following places: within a NOTICE text file distributed
       as part of the Derivative Works; within the Source form or
       documentation, if provided along with the Derivative Works; or,
       within a display generated by the Derivative Works, if and
       wherever such third-party notices normally appear. The contents
       of the NOTICE file are for informational purposes only and
       do not modify the License. You may add Your own attribution
       notices within Derivative Works that You distribute, alongside
       or as an addendum to the NOTICE text from the Work, provided
       that such additional attribution notices cannot be construed
       as modifying the License.

   You may add Your own copyright statement to Your modifications and
   may provide additional or different license terms and conditions
   for use, reproduction, or distribution of Your modifications, or
   for any such Derivative Works as a whole, provided Your use,
   reproduction, and distribution of the Work otherwise complies with
   the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
```

(continues on next page)

(continued from previous page)

```
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
   implied, including, without limitation, any warranties or conditions
   of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
   PARTICULAR PURPOSE. You are solely responsible for determining the
   appropriateness of using or redistributing the Work and assume any
   risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "{}"
   replaced with your own identifying information. (Don't include
   the brackets!)  The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
```

(continues on next page)

(continued from previous page)

```
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

gopkg.in/yaml.v3

License: MIT

```
This project is covered by two different licenses: MIT and Apache.

#### MIT License ####

The following files were ported to Go from C files of libyaml, and thus
are still covered by their original MIT license, with the additional
copyright staring in 2011 when the project was ported over:

    apic.go emitterc.go parserc.go readerc.go scannerc.go
    writerc.go yamlh.go yamlprivateh.go

Copyright (c) 2006-2010 Kirill Simonov
Copyright (c) 2006-2011 Kirill Simonov

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

### Apache License ###

All the remaining project files are covered by the Apache license:

Copyright (c) 2011-2019 Canonical Ltd
```

(continues on next page)

(continued from previous page)

```
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

gotest.tools/v3

License: Apache-2.0

```
Copyright 2018 gotest.tools authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

gotest.tools/v3/internal/difflib

License: BSD-3-Clause

```
Copyright (c) 2013, Patrick Mezard
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
    The names of its contributors may not be used to endorse or promote
products derived from this software without specific prior written
```

(continues on next page)

(continued from previous page)

```
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

mvdan.cc/sh/v3

License: BSD-3-Clause

```
Copyright (c) 2016, Daniel Martí. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of the copyright holder nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

sigs.k8s.io/yaml

License: MIT

```
The MIT License (MIT)

Copyright (c) 2014 Sam Ghods

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.


Copyright (c) 2012 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

   * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
```

(continues on next page)

(continued from previous page)

```
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```