

k8s.gcr.io/etcd docker image

Provides docker images containing etcd and etcdctl binaries for multiple etcd version as well as a migration operator utility for upgrading and downgrading etcd—it's data directory in particular—to a target version.

Versioning Each `k8s.gcr.io/etcd` docker image is tagged with an version string of the form `<etcd-version>-<image-revision>`, e.g. `3.0.17-0`. The etcd version is the SemVer of latest etcd version available in the image. The image revision distinguishes between docker images with the same latest etcd version but changes (bug fixes and backward compatible improvements) to the migration utility bundled with the image.

In addition to the latest etcd version, each `k8s.gcr.io/etcd` image contains etcd and etcdctl binaries for older versions of etcd. These are used by the migration operator utility when performing downgrades and multi-step upgrades, but can also be used as the etcd target version.

Usage Always run `/usr/local/bin/migrate` (or the `/usr/local/bin/migrate-if-needed.sh` wrapper script) before starting the etcd server. On Windows, run `C:\bin\migrate.exe` (or the `C:\bin\migrate-if-needed.bat` wrapper script).

`migrate` writes a `version.txt` file to track the “current” version of etcd that was used to persist data to disk. A “target” version may also be provided by the `TARGET_STORAGE` (e.g. “etcd3”) and `TARGET_VERSION` (e.g. “3.4.13”) environment variables. If the persisted version differs from the target version, `migrate-if-needed.sh` will migrate the data from the current to the target version.

Upgrades to any target version are supported. The data will be automatically upgraded in steps to each minor version until the target version is reached.

Downgrades to the previous minor version of the 3.x series is supported.

Permissions By default, `migrate` will write data directory files with default permissions according to the umask it is run with. When run in the published `k8s.gcr.io/etcd` images the default umask is 0022 which will result in 0755 directory permissions and 0644 file permissions.

Cross building For `amd64`, official `etcd` and `etcdctl` binaries are downloaded from Github to maintain official support. For other architectures, `etcd` is cross-compiled from source. Arch-specific `busybox` images serve as base images.

Windows images can be built on Linux nodes due to `docker buildx`, but they will only be created and pushed when using the `all-push` make target.

How to release First, update ETCD_VERSION and REVISION in the Makefile.

Next, build and test the image:

```
$ make build test
```

Last, build and push the docker images for all supported architectures.

```
# Build images for all the architecture and push the manifest image as well
```

```
$ make all-push
```

```
# Build images for all the architecture
```

```
$ make all-build
```

```
# Build image for target architecture(default=amd64)
```

```
$ make build ARCH=ppc64le
```

If you don't want to push the images, run `make` or `make build` instead