

Heketi Client Getting Started

This section describes the installation of Heketi CLI client on RHEL.

It is assumed the Heketi server is already installed and running and its URL, admin user and admin password are known.

The Heketi server and client are only used for administration of the GlusterFS storage. At run time the clients using the storage do not use Heketi. The clients use the GlusterFS client and work directly with the GlusterFS servers.

See [Managing Volumes using Heketi](#) for Red Hat documentation on the Heketi installation and configuration.

- A yum repository needs to be configured to get the Heketi RPMs.
Here is a sample yum `glusterfs.repo` file that needs to be created in `/etc/yum.repos.d/`.

```
[Gluster_4.0]
name=Gluster 3.13
baseurl=http://mirror.centos.org/centos/7/storage/$basearch/gluster-4.0/
gpgcheck=0
enabled=1
```

- It is assumed a yum repo has been configured that points to a GlusterFS repository.
- Install Heketi CLI:

```
> yum -y install heketi-client
```

To get general usage help with the heketi-cli, use `heketi-cli --help`.

To get more specific help use `heketi-cli command --help` where `command` is one of the heketi-cli commands. (The "commands" are not verbs but rather an object type or class, e.g., `topology`, `cluster`, `volume`, `node` and `device`.)

To get specific help for the "commands" associated with a given object type use `heketi-cli type command --help`, for example `heketi-cli topology load --help`.

When working with heketi-cli it is very convenient to export values for the following

environment variables:

```
> export HEKETI_CLI_SERVER=http://<heketi_server_host>:8080
> export HEKETI_CLI_USER=<heketi_admin>
> export HEKETI_CLI_KEY=<heketi_admin_password>
```

Substitute your Heketi server host name or IP address for `<heketi_server_host>` . Substitute the correct port if the Heketi server is not listening on `8080` . Likewise substitute the actual admin user name and password.

NOTE: When providing the Heketi server URL, be sure not to include a trailing slash on the URL. So for example `http://<heketi_server_host>:8080/` will lead to problems. The trailing slash causes an issue for a volume create operation, for example.

You may want to create a shell file named something like `configHeketiCLI.sh` and put the export statements there and then you can `source` the file. Be careful to protect the file as it has user and password in it.

Initializing the topology

The topology should have already been defined.

- Use the `heketi-cli` to load the `topology.json` file.

```
> heketi-cli topology load --json=topology.json
```

Obviously, the above command is assumed to have been run from the directory where the `topology.json` file is located.

Once the topology has successfully loaded you can use `heketi-cli topology info` to see information about the topology.

At this point you are ready to create mountable volumes that can be used by the ICP master nodes for shared storage.

Things that can go wrong with Heketi CLI

Error: Unknown user

This is caused by not having a `--user` argument or `HEKETI_CLI_USER` set.

Error: signature is invalid

When you run an `heketi-cli` command, you need to provide a user and "key" (aka password or secret). One way to provide the key is to set the environment variable, `HEKETI_CLI_KEY`. You can also pass it in on the `heketi-cli` command line with the `--secret` option. (See `heketi-cli --help` usage info.)

If the password/secret you provide does not match up with the user and key in the `heketi.json` file in current use by the Heketi service, then you will get the "signature is invalid" error.

NOTE: If you change anything in the `heketi.json` file you need to restart the Heketi service (`systemctl restart heketi`) to pick up the changes.

Using `heketi-cli` to create persistent volumes

NOTE: The `heketi-cli` commands in this section assume that the following environment variables have been set appropriately:

- `HEKETI_CLI_SERVER`
- `HEKETI_CLI_USER`
- `HEKETI_CLI_KEY`
- Get the `glusterfs` cluster ID. The topology info will have the cluster ID.

```
heketi-cli topology info
```

NOTE: The `heketi` doc recommends not providing a name for the created persistent volumes. That is debatable. If you have a good naming convention that ensures uniqueness, then using a descriptive name seems like a good idea rather than using the default names that are generated strings which have no descriptive value.)

- Create a 10GB volume with a default name:

```
> heketi-cli volume create --size=10 --cluster=<CLUSTER_ID>
```

The `--name <your_volume_name>` can be included with the above to provide a volume name rather than allow Heketi to create a generated volume name.

Resulting information for the volume created:

```
Name: vol_de785f066ce0e0ce86c39c5fb920682c
Size: 10
Volume Id: de785f066ce0e0ce86c39c5fb920682c
Cluster Id: 042e3eb4b386b086c17d9d947e8ba885
Mount: 172.16.20.17:vol_de785f066ce0e0ce86c39c5fb920682c
Mount Options: backup-volfile-servers=172.16.20.15,172.16.20.16
Durability Type: replicate
Distributed+Replica: 3
```

- Make note of the mountable volume host and name. That is what is used in the mount command and the entry in `/etc/fstab` on the VMs that will be using the volume. In the above example it is: `172.16.20.17:vol_de785f066ce0e0ce86c39c5fb920682c` You can replace the IP address with an actual hostname if you have an entry for the GlusterFS servers in DNS or `/etc/hosts` on the VMs that will be using the volume.
- Mount the volume on each node (Note the colon used to separate the backup servers rather than a comma.) In these examples, pseudo host names are used for each of 3 GlusterFS servers.

```
> mount -t glusterfs -o backup-volfile-servers=gluster01.xxx.yyy:gluster02.yyy.zzz
```



- Add a line to `/etc/fstab` :

```
gluster03.xxx.yyy:vol_de785f066ce0e0ce86c39c5fb920682c /var/lib/icp/audit glusterfs
```

