Managing Ceph Authentication

Objectives

After completing this section, you should be able to describe Cephx and configure user authentication and authorization for Ceph clients.

Authenticating Users

Red Hat Ceph Storage uses the `cephx` protocol to authorize communication between clients, applications, and daemons in the cluster. The `cephx` protocol is based on shared secret keys.

The installation process enables `cephx` by default, so that the cluster requires user authentication and authorization by all client applications.

Ceph uses user accounts for several purposes:

- For internal communication between Ceph daemons.
- For client applications accessing the cluster through the `librados` library.
- For cluster administrators.

Accounts used by Ceph daemons have names that match their associated daemon: `osd.1` or `mgr.serverc` and are created during the installation.

Accounts used by client applications that use `librados` have names with the `client.` prefix. For example, when integrating OpenStack with Ceph, it is common to create a dedicated `client.openstack` user account. For the Ceph Object Gateway, the installation creates a dedicated `client.rgw.`*`hostname`* user account. Developers creating custom software on top of `librados` should create dedicated accounts with appropriate capabilities.

Administrator account names also have the `client.` prefix. These accounts are used when running commands such as `ceph` and `rados`. The installer creates the superuser account, `client.admin`, with capabilities that allow the account to access everything and to modify the cluster configuration. Ceph uses the `client.admin` account to run administrative commands, unless you explicitly specify a user name with the `--name` or `--id` options.

You can set the `CEPH_ARGS` environment variable to define parameters such as the cluster name or the ID of the user.

```
[ceph: root@node /]# export CEPH_ARGS="--id cephuser"
```

End users of a Ceph-aware application do not have an account on the Ceph cluster. Rather, they access the application, which then accesses Ceph on their behalf. From the Ceph point of view, the application is the client. The application might provide its own user authentication through other mechanisms.

Figure 4.12 provides an overview of how an application can provide its own user authentication.