```
[root@node ~]# mount.ceph server.example.com:/ /mnt/mycephfs
[root@node ~]# mkdir /mnt/mycephfs/.snap/snap-name
```

Authorize the client to make snapshots for the CephFS file system with the s option.

```
[ceph: root@target /]# ceph fs authorize fs-name client path rws
```

To restore a file, copy it from the snapshot directory to another normal directory.

```
[root@node ~]# cp -a .snap/snap-name/file-name .
```

To fully restore a snapshot from the `.snap` directory tree, replace the normal entries with copies from the chosen snapshot.

```
[root@node ~]# rm -rf *
[root@node ~]# cp -a .snap/snap-name/* .
```

To discard a snapshot, remove the corresponding directory in `.snap`. The `rmdir` command succeeds even if the snapshot directory is not empty, without needing to use a recursive `rm` command.

```
[root@node ~]# rmdir .snap/snap-name
```

Scheduling Snapshots

You can use CephFS to schedule snapshots. The `snap_schedule` module manages the scheduled snapshots. You can use this module to create and delete snapshot schedules. Snapshot schedule information is stored in the CephFS metadata pool.

To create a snapshot schedule, first enable the `snap_schedule` module on the MGR node.

```
[ceph: root@server /]# ceph mgr module enable snap_schedule
```

Then, add the new snapshot schedule.

```
[ceph: root@server /]# ceph fs snap-schedule add fs-path time-period [start-time]
```

If an earlier version than Python 3.7 is installed, then the `start-time` string must use the format `%Y-%m-%dT%H:%M:%S`. For Python version 3.7 or later, you can use more flexible date parsing. For example, to create a snapshot schedule to create a snapshot for the `/volume` folder every hour, you can use the `ceph fs snap-schedule add` command.

```
[ceph: root@server /]# ceph fs snap-schedule add /volume 1h
Schedule set for path /volume
```

On the client node, review the snapshots in the `.snap` folder on your mounted CephFS: