ADDING/REMOVING OSDS

When you have a cluster up and running, you may add OSDs or remove OSDs from the cluster at runtime.

ADDING OSDS

When you want to expand a cluster, you may add an OSD at runtime. With Ceph, an OSD is generally one Ceph `ceph-osd` daemon for one storage drive within a host machine. If your host has multiple storage drives, you may map one `ceph-osd` daemon for each drive.

Generally, it's a good idea to check the capacity of your cluster to see if you are reaching the upper end of its capacity. As your cluster reaches its `near full` ratio, you should add one or more OSDs to expand your cluster's capacity.

Warning: Do not let your cluster reach its `full ratio` before adding an OSD. OSD failures that occur after the cluster reaches its `near full` ratio may cause the cluster to exceed its `full ratio`.

DEPLOY YOUR HARDWARE

If you are adding a new host when adding a new OSD, see Hardware Recommendations for details on minimum recommendations for OSD hardware. To add an OSD host to your cluster, first make sure you have an up-to-date version of Linux installed, and you have made some initial preparations for your storage drives. See Filesystem Recommendations for details.

Add your OSD host to a rack in your cluster, connect it to the network and ensure that it has network connectivity. See the Network Configuration Reference for details.

INSTALL THE REQUIRED SOFTWARE

For manually deployed clusters, you must install Ceph packages manually. See Installing Ceph (Manual) for details. You should configure SSH to a user with password-less authentication and root permissions.

ADDING AN OSD (MANUAL)

This procedure sets up a `ceph-osd` daemon, configures it to use one drive, and configures the cluster to distribute data to the OSD. If your host has multiple drives, you may add an OSD for each drive by repeating this procedure.

To add an OSD, create a data directory for it, mount a drive to that directory, add the OSD to the cluster, and then add it to the CRUSH map.

When you add the OSD to the CRUSH map, consider the weight you give to the new OSD. Hard drive capacity grows 40% per year, so newer OSD hosts may have larger hard drives than older hosts in the cluster (i.e., they may have greater weight).

Tip: Ceph prefers uniform hardware across pools. If you are adding drives of dissimilar size, you can adjust their weights. However, for best performance, consider a CRUSH hierarchy with drives of the same type/size.

1. Create the OSD. If no UUID is given, it will be set automatically when the OSD starts up. The following command will output the OSD number, which you will need for subsequent steps.

   ```
   ceph osd create [{uuid} [{id}]]
   ```

   If the optional parameter {id} is given it will be used as the OSD id. Note, in this case the command may fail if the number is already in use.

   Warning: In general, explicitly specifying {id} is not recommended. IDs are allocated as an array, and skipping entries consumes some extra memory. This can become significant if there are large gaps and/or clusters are large. If {id} is not specified, the smallest available is used.

2. Create the default directory on your new OSD.

```
ssh {new-osd-host}
sudo mkdir /var/lib/ceph/osd/ceph-{osd-number}
```

3. If the OSD is for a drive other than the OS drive, prepare it for use with Ceph, and mount it to the directory you just created:

   ```
   ssh {new-osd-host}
   sudo mkfs -t {fstype} /dev/{drive}
   sudo mount -o user_xattr /dev/{hdd} /var/lib/ceph/osd/ceph-{osd-number}
   ```

4. Initialize the OSD data directory.

   ```
   ssh {new-osd-host}
   ceph-osd -i {osd-num} --mkfs --mkkey
   ```

   The directory must be empty before you can run `ceph-osd`.

5. Register the OSD authentication key. The value of ceph for ceph-`{osd-num}` in the path is the `$cluster-$id`. If your cluster name differs from ceph, use your cluster name instead.:

   ```
   ceph auth add osd.{osd-num} osd 'allow *' mon 'allow rwx' -i /var/lib/ceph/osd/ceph-{osd-
   ```

6. Add the OSD to the CRUSH map so that the OSD can begin receiving data. The ceph `osd crush` add command allows you to add OSDs to the CRUSH hierarchy wherever you wish. If you specify at least one bucket, the command will place the OSD into the most specific bucket you specify, *and* it will move that bucket underneath any other buckets you specify. **Important:** If you specify only the root bucket, the command will attach the OSD directly to the root, but CRUSH rules expect OSDs to be inside of hosts.

   For Argonaut (v 0.48), execute the following:

   ```
   ceph osd crush add {id} {name} {weight}  [{bucket-type}={bucket-name} ...]
   ```

   For Bobtail (v 0.56) and later releases, execute the following:

   ```
   ceph osd crush add {id-or-name} {weight}  [{bucket-type}={bucket-name} ...]
   ```

   You may also decompile the CRUSH map, add the OSD to the device list, add the host as a bucket (if it’s not already in the CRUSH map), add the device as an item in the host, assign it a weight, recompile it and set it. See Add/Move an OSD for details.

Argonaut (v0.48) Best Practices

To limit impact on user I/O performance, add an OSD to the CRUSH map with an initial weight of `0`. Then, ramp up the CRUSH weight a little bit at a time. For example, to ramp by increments of `0.2`, start with:

```
ceph osd crush reweight {osd-id} .2
```

and allow migration to complete before reweighting to `0.4`, `0.6`, and so on until the desired CRUSH weight is reached.

To limit the impact of OSD failures, you can set:

```
mon osd down out interval = 0
```

which prevents down OSDs from automatically being marked out, and then ramp them down manually with:

```
ceph osd reweight {osd-num} .8
```

> Again, wait for the cluster to finish migrating data, and then adjust the weight further until you reach a weight of 0. Note that this problem prevents the cluster to automatically re-replicate data after a failure, so please ensure that sufficient monitoring is in place for an administrator to intervene promptly.
>
> Note that this practice will no longer be necessary in Bobtail and subsequent releases.

REPLACING AN OSD

When disks fail, or if an admnistrator wants to reprovision OSDs with a new backend, for instance, for switching from FileStore to BlueStore, OSDs need to be replaced. Unlike Removing the OSD, replaced OSD's id and CRUSH map entry need to be keep intact after the OSD is destroyed for replacement.

1. Destroy the OSD first:

   ```
   ceph osd destroy {id} --yes-i-really-mean-it
   ```

2. Zap a disk for the new OSD, if the disk was used before for other purposes. It's not necessary for a new disk:

   ```
   ceph-volume lvm zap /dev/sdX
   ```

3. Prepare the disk for replacement by using the previously destroyed OSD id:

   ```
   ceph-volume lvm  prepare --osd-id {id} --data /dev/sdX
   ```

4. And activate the OSD:

   ```
   ceph-volume lvm activate {id} {fsid}
   ```

Alternatively, instead of preparing and activating, the device can be recreated in one call, like:

```
ceph-volume lvm create --osd-id {id} --data /dev/sdX
```

STARTING THE OSD

After you add an OSD to Ceph, the OSD is in your configuration. However, it is not yet running. The OSD is down and `in`. You must start your new OSD before it can begin receiving data. You may use `service` ceph from your admin host or start the OSD from its host machine.

For Ubuntu Trusty use Upstart.

```
sudo start ceph-osd id={osd-num}
```

For all other distros use systemd.

```
sudo systemctl start ceph-osd@{osd-num}
```

Once you start your OSD, it is up and `in`.

OBSERVE THE DATA MIGRATION

Once you have added your new OSD to the CRUSH map, Ceph will begin rebalancing the server by migrating placement groups to your new OSD. You can observe this process with the ceph tool.

```
ceph -w
```

You should see the placement group states change from `active+clean` to `active, some degraded objects`, and finally `active+clean` when migration completes. (Control-c to exit.)

REMOVING OSDS (MANUAL)

When you want to reduce the size of a cluster or replace hardware, you may remove an OSD at runtime. With Ceph, an OSD is generally one Ceph `ceph-osd` daemon for one storage drive within a host machine. If your host has multiple storage drives, you may need to remove one `ceph-osd` daemon for each drive. Generally, it's a good idea to check the capacity of your cluster to see if you are reaching the upper end of its capacity. Ensure that when you remove an OSD that your cluster is not at its `near full` ratio.

Warning: Do not let your cluster reach its `full ratio` when removing an OSD. Removing OSDs could cause the cluster to reach or exceed its `full ratio`.

TAKE THE OSD OUT OF THE CLUSTER

Before you remove an OSD, it is usually `up` and `in`. You need to take it out of the cluster so that Ceph can begin rebalancing and copying its data to other OSDs.

```
ceph osd out {osd-num}
```

OBSERVE THE DATA MIGRATION

Once you have taken your OSD `out` of the cluster, Ceph will begin rebalancing the cluster by migrating placement groups out of the OSD you removed. You can observe this process with the ceph tool.

```
ceph -w
```

You should see the placement group states change from `active+clean` to `active, some degraded objects`, and finally `active+clean` when migration completes. (Control-c to exit.)

Note: Sometimes, typically in a "small" cluster with few hosts (for instance with a small testing cluster), the fact to take out the OSD can spawn a CRUSH corner case where some PGs remain stuck in the `active+remapped` state. If you are in this case, you should mark the OSD `in` with:

```
ceph osd in {osd-num}
```

to come back to the initial state and then, instead of marking `out` the OSD, set its weight to 0 with:

```
ceph osd crush reweight osd.{osd-num} 0
```

After that, you can observe the data migration which should come to its end. The difference between marking `out` the OSD and reweighting it to 0 is that in the first case the weight of the bucket which contains the OSD is not changed whereas in the second case the weight of the bucket is updated (and decreased of the OSD weight). The reweight command could be sometimes favoured in the case of a "small" cluster.

STOPPING THE OSD

After you take an OSD out of the cluster, it may still be running. That is, the OSD may be `up` and `out`. You must stop your OSD before you remove it from the configuration.

```
ssh {osd-host}
sudo systemctl stop ceph-osd@{osd-num}
```

Once you stop your OSD, it is `down`.

REMOVING THE OSD

This procedure removes an OSD from a cluster map, removes its authentication key, removes the OSD from the OSD map, and removes the OSD from the `ceph.conf` file. If your host has multiple drives, you may need to remove an OSD for each drive by repeating this procedure.

1. Let the cluster forget the OSD first. This step removes the OSD from the CRUSH map, removes its authentication key. And it is removed from the OSD map as well. Please note the purge subcommand is introduced in Luminous, for older versions, please see below

   ```
   ceph osd purge {id} --yes-i-really-mean-it
   ```

2. Navigate to the host where you keep the master copy of the cluster's `ceph.conf` file.

   ```
   ssh {admin-host}
   cd /etc/ceph
   vim ceph.conf
   ```

3. Remove the OSD entry from your `ceph.conf` file (if it exists).

   ```
   [osd.1]
           host = {hostname}
   ```

4. From the host where you keep the master copy of the cluster's `ceph.conf` file, copy the updated `ceph.conf` file to the `/etc/ceph` directory of other hosts in your cluster.

If your Ceph cluster is older than Luminous, instead of using `ceph osd purge`, you need to perform this step manually:

1. Remove the OSD from the CRUSH map so that it no longer receives data. You may also decompile the CRUSH map, remove the OSD from the device list, remove the device as an item in the host bucket or remove the host bucket (if it's in the CRUSH map and you intend to remove the host), recompile the map and set it. See Remove an OSD for details.

   ```
   ceph osd crush remove {name}
   ```

2. Remove the OSD authentication key.

   ```
   ceph auth del osd.{osd-num}
   ```

   The value of ceph for `ceph-{osd-num}` in the path is the `$cluster-$id`. If your cluster name differs from ceph, use your cluster name instead.

3. Remove the OSD.

   ```
   ceph osd rm {osd-num}
   #for example
   ceph osd rm 1
   ```