

When Ceph starts, it uses the `ceph-crush-location` utility to automatically verify that each OSD is in the correct CRUSH location. If the OSD is not in the expected location in the CRUSH map, it is automatically moved. By default, this is `root=default host=hostname`.

You can replace the `ceph-crush-location` utility with your own script to change where OSDs are placed in the CRUSH map. To do this, specify the `crush_location_hook` parameter in the `/etc/ceph/ceph.conf` configuration file.

```
...output omitted...
[osd]
crush_location_hook = /path/to/your/script
...output omitted...
```

Ceph executes the script with these arguments: `--cluster cluster-name --id osd-id --type osd`. The script must print the location as a single line on its standard output. The upstream Ceph documentation has an example of a custom script that assumes each system has an `/etc/rack` file containing the name of its rack:

```
#!/bin/sh
echo "root=default rack=$(cat /etc/rack) host=$(hostname -s)"
```

You can set the `crush_location` parameter in the `/etc/ceph/ceph.conf` configuration file to redefine the location for particular OSDs. For example, to set the location for `osd.0` and `osd.1`, add the `crush_location` parameter inside their respective sections in that file:

```
[osd.0]
crush_location = root=default datacenter=DC1 rack=rackA1

[osd.1]
crush_location = root=default datacenter=DC1 rack=rackB1
```

Adding CRUSH Map Rules

This example creates a rule that Ceph can use for replicated pools:

```
[ceph: root@node /]# ceph osd crush rule create-replicated name root
\
failure-domain-type [class]
```

- `name` is the name of the rule.
- `root` is the starting node in the CRUSH map hierarchy.
- `failure-domain-type` is the bucket type for replication.
- `class` is the class of the devices to use, such as `ssd` or `hdd`. This parameter is optional.

The following example creates the new `inDC2` rule to store replicas in the DC2 data center, and distributes the replicas across racks:

```
[ceph: root@node /]# ceph osd crush rule create-replicated inDC2 DC2 rack
[ceph: root@node /]# ceph osd crush rule ls
replicated_rule
erasure-code
inDC2
```