Red Hat OpenStack Platform 16.2

Recommendations for Large Deployments

Hardware requirements and configuration for deploying Red Hat OpenStack Platform at scale

Last Updated: 2024-06-18

Red Hat OpenStack Platform 16.2 Recommendations for Large Deployments

Hardware requirements and configuration for deploying Red Hat OpenStack Platform at scale

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Abstract

This guide contains several recommendations for deploying Red Hat OpenStack Platform at scale. These recommendations include hardware recommendations, undercloud tuning, and overcloud configuration.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message .

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Tell us how we can make it better.

Providing documentation feedback in Jira

Use the Create Issue form to provide feedback on the documentation. The Jira issue will be created in the Red Hat OpenStack Platform Jira project, where you can track the progress of your feedback.

1. Ensure that you are logged in to Jira. If you do not have a Jira account, create an account to submit feedback.

2. Click the following link to open a the **Create Issue** page: Create Issue

3. Complete the **Summary** and **Description** fields. In the **Description** field, include the documentation URL, chapter or section number, and a detailed description of the issue. Do not modify any other fields in the form.

4. Click **Create**.

CHAPTER 1. RECOMMENDATIONS FOR LARGE DEPLOYMENTS

Use the following undercloud and overcloud recommendations, specifications, and configuration for deploying a large Red Hat OpenStack Platform (RHOSP) environment. RHOSP 16.2 deployments that include more than 100 overcloud nodes are considered large environments. Red Hat has tested and validated optimum performance on a large scale environment of 700 overcloud nodes using RHOSP 16.2 without using minion.

For DCN-based deployments, the number of nodes from the central and edge sites can be very high. For recommendations about DCN deployments, contact Red Hat Global Support Services.

CHAPTER 2. RECOMMENDED SPECIFICATIONS FOR YOUR LARGE RED HAT OPENSTACK DEPLOYMENT

You can use the provided recommendations to scale your large cluster deployment.

The values in the following procedures are based on testing that the Red Hat OpenStack Platform Performance & Scale Team performed and can vary according to individual environments. For more information, see Scaling Red Hat OpenStack Platform 16.1 to more than 700 nodes .

2.1. UNDERCLOUD SYSTEM REQUIREMENTS

For best performance, install the undercloud node on a physical server. However, if you use a virtualized undercloud node, ensure that the virtual machine has enough resources similar to a physical machine described in the following table.

Table 2.1. Recommended specifications for the undercloud node

System requirement	Description
Counts	1
CPUs	32 cores, 64 threads
Disk	500 GB root disk (1x SSD or 2x hard drives with 7200RPM; RAID 1) 500 GB disk for Object Storage (swift) (1x SSD or 2x hard drives with 7200RPM; RAID 1)
Memory	256 GB
Network	25 Gbps network interfaces or 10 Gbps network interfaces

2.2. OVERCLOUD CONTROLLER NODES SYSTEM REQUIREMENTS

All control plane services must run on exactly 3 nodes. Typically, all control plane services are deployed across 3 Controller nodes.

Scaling controller services

To increase the resources available for controller services, you can scale these services to additional nodes. For example, you can deploy the **db** or **messaging** controller services on dedicated nodes to reduce the load on the Controller nodes.

To scale controller services, use composable roles to define the set of services that you want to scale. When you use composable roles, each service must run on exactly 3 additional dedicated nodes and the total number of nodes in the control plane must be odd to maintain Pacemaker quorum.

The control plane in this example consists of the following 9 nodes:

- 3 Controller nodes

- 3 Database nodes
- 3 Messaging nodes

For more information, see Composable services and custom roles in *Advanced Overcloud Customization*.

For questions about scaling controller services with composable roles, contact Red Hat Global Professional Services.

Storage considerations

Include sufficient storage when you plan Controller nodes in your overcloud deployment. OpenStack Telemetry Metrics (gnocchi) and OpenStack Image service (glance) services are I/O intensive. Use Ceph Storage and the Image service for telemetry because the overcloud moves the I/O load to the Ceph OSD servers.

If your deployment does not include Ceph storage, use a dedicated disk or node for Object Storage (swift) that Telemetry Metrics (gnocchi) and Image (glance) services can use. If you use Object Storage on Controller nodes, use an NVMe device separate from the root disk to reduce disk utilization during object data storage.

Extensive concurrent operations to the Block Storage service (cinder) that upload volumes to the Image Storage service (glance) as images puts considerable IO load on the controller disk. You can use SSD disks to provide a higher throughput.

CPU considerations

The number of API calls, AMQP messages, and database queries that the Controller nodes receive influences the CPU memory consumption on the Controller nodes. The ability of each Red Hat OpenStack Platform (RHOSP) component to concurrently process and perform tasks is also limited by the number of worker threads that are configured for each of the individual RHOSP components. The number of worker threads for components that RHOSP director configures on a Controller is limited by the CPU count.

The following specifications are recommended for large scale environments with more than 700 nodes when you use Ceph Storage nodes in your deployment:

Table 2.2. Recommended specifications for Controller nodes when you use Ceph Storage nodes

System requirement	Description
Counts	3 Controller nodes with controller services contained within the Controller role. Optionally, to scale controller services on dedicated nodes, use composable services. For more information, see Composable services and customer roles in *Advanced Overcloud Customization*.
CPUs	2 sockets each with 32 cores, 64 threads

System requirement	Description
Disk	500GB root disk (1x SSD or 2x hard drives with 7200RPM; RAID 1) 500GB dedicated disk for Swift (1x SSD or 1x NVMe) Optional: 500GB disk for image caching (1x SSD or 2x hard drives with 7200RPM; RAID 1)
Memory	384GB
Network	25 Gbps network interfaces or 10 Gbps network interfaces. If you use 10 Gbps network interfaces, use network bonding to create two bonds: • Provisioning (bond0 - mode4); Internal API (bond0 - mode4); Project (bond0 - mode4) • Storage (bond1 - mode4); Storage management (bond1 - mode4)

The following specifications are recommended for large scale environments with more than 700 nodes when you do not use Ceph Storage nodes in your deployment:

Table 2.3. Recommended specifications for Controller nodes when you do not use Ceph Storage nodes

System requirement	Description
Counts	3 Controller nodes with controller services contained within the Controller role. Optionally, to scale controller services on dedicated nodes, use composable services. For more information, see Composable services and customer roles in *Advanced Overcloud Customization*.
CPUs	2 sockets each with 32 cores, 64 threads
Disk	500GB root disk (1x SSD) 500GB dedicated disk for Swift (1x SSD or 1x NVMe) Optional: 500GB disk for image caching (1x SSD or 2x hard drives with 7200RPM; RAID 1)
Memory	384GB

System requirement	Description
Network	25 Gbps network interfaces or 10 Gbps network interfaces. If you use 10 Gbps network interfaces, use network bonding to create two bonds: • Provisioning (bond0 - mode4); Internal API (bond0 - mode4); Project (bond0 - mode4) • Storage (bond1 - mode4); Storage management (bond1 - mode4)

2.3. OVERCLOUD COMPUTE NODES SYSTEM REQUIREMENTS

When you plan your overcloud deployment, review the recommended system requirements for Compute nodes.

Table 2.4. Recommended specifications for Compute nodes

System requirement	Description
Counts	Red Hat has tested a scale of 700 nodes with various composable compute roles.
CPUs	2 sockets each with 12 cores, 24 threads
Disk	500GB root disk (1x SSD or 2x hard drives with 7200RPM; RAID 1)
Memory	128GB (64GB per NUMA node); 2GB is reserved for the host out by default. With Distributed Virtual Routing, increase the reserved RAM to 5GB.
Network	25 Gbps network interfaces or 10 Gbps network interfaces. If you use 10 Gbps network interfaces, use network bonding to create two bonds: • Provisioning (bond0 - mode4); Internal API (bond0 - mode4); Project (bond0 - mode4) • Storage (bond1 - mode4)

2.4. RED HAT CEPH STORAGE NODES SYSTEM REQUIREMENTS

When you plan your overcloud deployment, review the following recommended system requirements for Ceph storage nodes.

Table 2.5. Recommended specifications for Ceph Storage nodes

System requirement	Description
Counts	You must have a minimum of 5 nodes with three-way replication. With all-flash configuration, you must have a minimum of 3 nodes with two-way replication.
CPUs	1 Intel Broadwell CPU core per OSD to support storage I/O requirements. If you are using a light I/O workload, you might not need Ceph to run at the speed of your block devices. For example, for some NFV applications, Ceph supplies data durability, high availability, and low latency but throughput is not a target, so it is acceptable to supply less CPU power.
Memory	Ensure that you have 5 GB RAM per OSD. This is required for caching OSD data and metadata to optimize performance, not just for the OSD process memory. For hyper-converged infrastructure (HCI) environments, calculate the required memory in conjunction with the Compute node specifications.
Network	Ensure that the network capacity in MB/s is higher than the total MB/s capacity of the Ceph devices to support workloads that use a large I/O transfer size. Use a cluster network to lower write latency by shifting inter-OSD traffic onto a separate set of physical network ports. To do this in Red Hat OpenStack Platform, configure separate VLANs for networks and assign the VLANs to separate physical network interfaces.
Disk	Use Solid-State Drive (SSD) disks for the bluestore **block.db** partition to reduce I/O contention on hard disk drives (HDD), which increases the speed of write IOPS, but SSDs have zero effect on read input/output operations per second. If you use SATA/SAS SSD journals, you typically need a ratio of SSD:HDD of 1:5. If you use NVM SSD journals, you can typically use a SSD:HDD ratio of 1:10 or 1:15 if the workload is read-mostly. However, if this ratio is too high, the SSD journal device failure can affect the OSDs.

For more information about hardware prerequisites for Ceph nodes, see General principles for selecting hardware in the Red Hat Storage 4 *Hardware Guide*.

For more information about deployment configuration for Ceph nodes, see Deploying an overcloud with containerized Red Hat Ceph.

For more information about changing the storage replication number, see Pools, placement groups, and CRUSH Configuration Reference in the *Red Hat Ceph Storage Configuration Guide* .

CHAPTER 3. RED HAT OPENSTACK DEPLOYMENT BEST PRACTICES

Review the following best practices when you plan and prepare to deploy OpenStack. You can apply one or more of these practices in your environment.

3.1. RED HAT OPENSTACK DEPLOYMENT PREPARATION

Before you deploy Red Hat OpenStack Platform (RHOSP), review the following list of deployment preparation tasks. You can apply one or more of the deployment preparation tasks in your environment:

Set a subnet range for introspection to accommodate the maximum overcloud nodes for which you want to perform introspection at a time

When you use director to deploy and configure RHOSP, use CIDR notations for the control plane network to accommodate all overcloud nodes that you add now or in the future.

Set the root password on your overcloud image to allow console access to the overcloud image

Use the console to troubleshoot failed deployments when networking is set incorrectly. For more information, see Installing virt-customize to the director and Setting the Root Password in the *Partner Integration Guide*. Adhere to the information security policies of your organization for password management when you implement this recommendation.
Alternatively, you can use the **userdata_root_password.yaml** template to configure the root password by using the **NodeUserData** parameter. You can find the template in **/usr/share/openstack-tripleo-heat-templates/firstboot/userdata_root_password.yaml**.

The following example uses the template to configure the **NodeUserData** parameter:

```
resource_registry:
  OS::TripleO::NodeUserData: /usr/share/openstack-tripleo-heat-templates/firstboot/userdata_root_password.yaml
parameter_defaults:
  NodeRootPassword: '<password>'
```

Use scheduler hints to assign hardware to a role

- Use scheduler hints to assign hardware to a role, such as **Controller**, **Compute**, **CephStorage**, and others. Scheduler hints provide easier identification of deployment issues that affect only a specific piece of hardware.
- Do not use profile tagging when you use scheduler hints.
- In performance testing, use identical hardware for specific roles to reduce variability in testing and performance results.
- For more information, see Assigning Specific Node IDs in the *Advanced Overcloud Customization Guide*.

Set the World Wide Name (WWN) as the root disk hint for each node to prevent nodes from using the wrong disk during deployment and booting

When nodes contain multiple disks, use the introspection data to set the WWN as the root disk hint for each node. This prevents the node from using the wrong disk during deployment and booting. For more information, see Defining the Root Disk for Multi-Disk Clusters in the *Director Installation and Usage* guide.

Enable the Bare Metal service (ironic) automated cleaning on nodes that have more than one disk

Use the Bare Metal service automated cleaning to erase metadata on nodes that have more than one disk and are likely to have multiple boot loaders. Nodes might become inconsistent with the boot disk due to the presence of multiple bootloaders on disks, which leads to node deployment failure when you attempt to pull the metadata that uses the wrong URL.
To enable the Bare Metal service automated cleaning, on the undercloud node, edit the **undercloud.conf** file and add the following line:

```
clean_nodes = true
```

Limit the number of nodes for Bare Metal (ironic) introspection

If you perform introspection on all nodes at the same time, failures might occur due to network constraints. Perform introspection on up to 50 nodes at a time.
Ensure that the **dhcp_start** and **dhcp_end** range in the **undercloud.conf** file is large enough for the number of nodes that you expect to have in the environment.

If there are insufficient available IPs, do not issue more than the size of the range. This limits the number of simultaneous introspection operations. To allow the introspection DHCP leases to expire, do not issue more IP addresses for a few minutes after the introspection completes.

Prepare Ceph for different types of configurations

The following list is a set of recommendations for different types of configurations:

- **All-flash OSD configuration**
  Each OSD requires additional CPUs according to the IOPS capacity of the device type, so Ceph IOPS are CPU-limited at a lower number of OSDs. This is true for NVM SSDs, which can have two orders of magnitude higher IOPS capacity than traditional HDDs. For SATA/SAS SSDs, expect one order of magnitude greater random IOPS/OSD than HDDs, but only about two to four times the sequential IOPS increase. You can supply less CPU resources to Ceph than Ceph needs for OSD devices.

- **Hyper Converged Infrastructure (HCI)**
  It is recommended to reserve at least half of your CPU capacity, memory, and network for the OpenStack Compute (nova) guests. Ensure that you have enough CPU capacity and memory to support both OpenStack Compute (nova) guests and Ceph Storage. Observe memory consumption because Ceph Storage memory consumption is not elastic. On a multi-CPUs socket system, limit Ceph CPU consumption with NUMA-pinning Ceph to a single socket. For example, use the **numactl -N 0 -p 0** command. Do not hard-pin Ceph memory consumption to 1 socket.

- **Latency-sensitive applications such as NFV**
  Place Ceph on the same CPU socket as the network card that Ceph uses and limit the network card interruptions to that CPU socket if possible, with a network application that runs on a different NUMA socket and network card.

  If you use dual bootloaders, use disk-by-path for the OSD map. This gives the user consistent deployments, unlike using the device name. The following snippet is an example of the **CephAnsibleDisksConfig** for a disk-by-path mapping.

  ```
  CephAnsibleDisksConfig:
    osd_scenario: non-collocated
    devices:
      - /dev/disk/by-path/pci-0000:03:00.0-scsi-0:2:0:0
      - /dev/disk/by-path/pci-0000:03:00.0-scsi-0:2:1:0
  ```

```
dedicated_devices:
  - /dev/nvme0n1
  - /dev/nvme0n1
journal_size: 512
```

3.2. RED HAT OPENSTACK DEPLOYMENT CONFIGURATION

Review the following list of recommendations for your Red Hat OpenStack Platform(RHOSP) deployment configuration:

Validate the heat templates with a small scale deployment

Deploy a small environment that consists of at least three Controllers, one Compute note, and three Ceph Storage nodes. You can use this configuration to ensure that all of your heat templates are correct.

Disable telemetry notifications on the undercloud

You can disable telemetry notifications on the undercloud for the following OpenStack services to decrease the RabbitMQ queue:

- Compute (nova)
- Networking (neutron)
- Orchestration (heat)
- Identity (keystone)

To disable the notifications, in the **/usr/share/openstack-tripleo-heat-templates/environments/disable-telemetry.yaml** template, set the notification driver setting to **noop**.

Limit the number of nodes that are provisioned at the same time

Fifty is the typical amount of servers that can fit within a average enterprise-level rack unit, therefore, you can deploy an average of one rack of nodes at one time.
To minimize the debugging necessary to diagnose issues with the deployment, deploy no more than 50 nodes at one time. However, if you want to deploy a higher number of nodes, Red Hat has successfully tested up to 100 nodes simultaneously.

To scale Compute nodes in batches, use the **openstack overcloud deploy** command with the **--limit** option. This can result in saved time and lower resource consumption on the undercloud.

> **NOTE**
>
> The **--limit** option is in Technology Preview.
>
> Use this option with a comma-separated list of tags from the config-download playbook to run the deployment with a specific set of config-download tasks.

Disable unused NICs

If the overcloud has any unused NICs during the deployment, you must define the unused interfaces in the NIC configuration templates and set the interfaces to **use_dhcp: false** and **defroute: false**.
If you do not define unused interfaces, there might be routing issues and IP allocation problems

during introspection and scaling operations. By default, the NICs set **BOOTPROTO=dhcp**, which means the unused overcloud NICs consume IP addresses that are needed for the PXE provisioning. This can reduce the pool of available IP addresses for your nodes.

Power off unused Bare Metal Provisioning (ironic) nodes

Ensure that you power off any unused Bare Metal Provisioning (ironic) nodes in maintenance mode. Red Hat has identified cases where nodes from previous deployments are left in maintenance mode in a powered on state. This can occur with Bare Metal automated cleaning, where a node that fails cleaning is set to maintenance mode. Bare Metal Provisioning does not track the power state of nodes in maintenance mode and incorrectly reports the power state as off. This can cause problems with ongoing deployments. When you redeploy after a failed deployment, ensure that you power off all unused nodes that use the power management device of the node.

3.3. TUNING THE UNDERCLOUD

Review this section when you plan to scale your RHOSP deployment and apply tuning to your default undercloud settings.

If you use the Telemetry service (ceilometer), improve the performance of the service

Because the Telemetry service is CPU-intensive, telemetry is not enabled by default in RHOSP 16.2. If you use want to use Telemetry, you can improve the performance of the service.
For more information, see Configuration recommendations for the Telemetry service in the *Deployment Recommendations for Specific Red Hat OpenStack Platform Services Guide* .

Separate the provisioning and configuration processes

- To create only the stack and associated RHOSP resources, you can run the deployment command with the **--stack-only** option.
- Red Hat recommends separating the stack and config-download steps when deploying more than 100 nodes.

Include any environment files that are required for your overcloud:

```
$ openstack overcloud deploy \
  --templates \
  -e <environment-file1.yaml> \
  -e <environment-file2.yaml> \
  ...
  --stack-only
```

- After you have provisioned the stack, you can enable the SSH access for the **tripleo-admin** user from the undercloud to the overcloud. The **config-download** process uses the **tripleo-admin** user to perform the Ansible based configuration:

  ```
  $ openstack overcloud admin authorize
  ```

- To disable the overcloud stack creation and run only the **config-download** workflow to apply the software configuration, you can run the deployment command with the **--config-download-only** option. Include any environment files that are required for your overcloud:

  ```
  $ openstack overcloud deploy \
   --templates \
  ```

```
-e <environment-file1.yaml> \
-e <environment-file2.yaml> \
 ...
--config-download-only
```

- To limit the **config-download** playbook execution to a specific node or set of nodes, you can use the **--limit** option.
- The **--limit** option can be used to separate nodes into different roles, to limit the number of nodes to deploy, or to separate nodes with a specific hardware type. For scale up operations, when you want to apply software configuration on the new nodes only, use the **--limit** option with the **--config-download-only** option.

```
$ openstack overcloud deploy \
--templates \
-e <environment-file1.yaml> \
-e <environment-file2.yaml> \
...
--config-download-only --config-download-timeout --limit <Undercloud>,<Controller>,
<Compute-1>,<Compute-2>
```

If you use the **--limit** option always include **<Controller>** and **<Undercloud>** in the list. Tasks that use the **external_deploy_steps** interface, for example all Ceph configurations, are only executed if the **<Undercloud>** is included in the options list. All **external_deploy_steps** tasks run on the undercloud.

For example, if you run a scale-up task to add a compute node that requires a connection to Ceph and you do not include **<Undercloud>** in the list, the Ceph configuration and **cephx** key files are missing, and the task fails. Do not use the **--skip-tags external_deploy_steps** option or the task fails.

3.4. TUNING THE OVERCLOUD

Review the following section when you plan to scale your Red Hat OpenStack Platform (RHOSP) deployment and apply tuning to your default overcloud settings:

Increase OVN OVSDB client probe intervals to prevent failover

Increase OVSDB client probe intervals for large RHOSP deployments. Pacemaker triggers a failover of the **ovn-dbs-bundle** when it does not get a response from OVN within the configured timeout. To increase the OVN OVSDB client probe intervals to 360 seconds, edit the **OVNDBSPacemakerTimeout** parameter in your heat templates:

```
OVNDBSPacemakerTimeout: 360
```

On each Compute and Controller node, the OVN controller periodically probes the OVN SBDB and if these requests timeout, the OVN controller resynchronizes. When multiple Compute and Controller nodes are loaded with requests to create resources, the default 60 seconds timeout values are not sufficient. To increase the OVN SBDB client probe intervals to 180 seconds, edit the **OVNOpenflowProbeInterval** parameter in your heat templates:

```
ControllerParameters:
  OVNRemoteProbeInterval: 180000
  OVNOpenflowProbeInterval: 180
```

NOTE

During RHOSP user and service triggered operations, due to resource constraints, such as CPU or memory resource constraints, multiple components can reach their configured timeout values. This can result in timeout request failures to the haproxy front end or back end, messaging timeout, db query-related failures, cluster instability, and so on. Benchmark your overcloud environment after initial deployment to help identify timeout-related bottlenecks.

Set a high interval between instance network information cache updates

The default interval between instance network information cache updates is 60 seconds: **heal_instance_info_cache_interval = 60**. To ensure that the system can manage the load that healing of the instance cache puts on the neutron server, modify the value of the **heal_instance_info_cache_interval** parameter for your environment. For example:

- For once every ten minutes, set **heal_instance_info_cache_interval = 600**.
- For once every hour, set **heal_instance_info_cache_interval = 3600**.

WARNING

Some third-party plugins return inconsistent API database responses under heavy loads. ML2/OVS and ML2/OVN backends do not experience an inconsistent API database response problem. If your deployment uses ML2/OVS or ML2/OVN, you can disable the **heal_instance_info_cache_interval** parameter by setting the value to **-1**.

For more information about the parameter, see nova in the *Configuration Reference* guide.

CHAPTER 4. DEBUGGING RECOMMENDATIONS AND KNOWN ISSUES

Review the following section for debugging suggestions that can help you troubleshoot your deployment.

4.1. KNOWN ISSUES

The following list outlines existing current limitations.

BZ#1857451 - Ansible forks value should have an upper limit and Current Calculation needs to change

By default, the Ansible playbooks in mistral are configured to use **10*CPU_COUNT** forks in the **ansible.cfg** file. When you do not use the **--limit** option to limit the Ansible execution to a specific node or set of nodes and the Ansible execution is set to run on all of the existing nodes, Ansible consumes almost 100% of memory utilisation.

4.2. INTROSPECTION DEBUGGING

Review the following list of recommendations when you debug introspection.

Check your introspection DHCP range and NICs in your undercloud.conf file

If any of these values are incorrect, fix them, and rerun the **openstack undercloud install** command.

Ensure that you do not try to introspect more than your DHCP range of nodes can allow

The DHCP lease for each node continues to be active for approximately two minutes after introspection finishes.

Ensure that target nodes are responsive

If all nodes fail introspection, ensure that you can ping target nodes over the native VLAN by using the configured NIC and that the out-of-band interface credentials and addresses are correct.

Check the introspection commands in the console

For debugging specific nodes, watch the console when the node boots and observe introspection commands to the node. If the node stops before it completes the PXE process, check the connectivity, IP allocation, and the network load. When a node exits the BIOS and boots the introspection image, failures are rare and almost exclusively related to connectivity issues. Ensure that the heartbeat from the introspection image is not interrupted on its way to the undercloud.

4.3. DEPLOYMENT DEBUGGING

Use the following recommendations when you debug a deployment.

Inspect the DHCP servers that provide addresses on the provisioning network

Any additional DHCP servers that supply addresses on the provisioning network can prevent Red Hat OpenStack Platform director from inspecting and provisioning machines.

- For DHCP or PXE introspection issues, enter the following command:

```
$ sudo tcpdump -i any port 67 or port 68 or port 69
```

- For DHCP or PXE deployment issues, enter the following command:

```
$ sudo ip netns exec qdhcp tcpdump -i <interface> port 67 or port 68 or port 69
```

Check the state of your failed or foreign disks

For failed or foreign disks, check the state of your disks to ensure that, according to the out-of-band management of the machine, the state of the failed or foreign disks is set to **Up**. Disks can exit the **Up** state during a deployment cycle and change the order that your disks appear in the base operating system.

Use the following commands to debug failed overcloud deployments

- **openstack stack failures list overcloud**
- **heat resource-list -n5 overcloud | grep -i fail**
- **less /var/lib/mistral/config-download-latest/ansible.log**

To review the output of the commands, log in to the node where the failure occurs and review the log files in **/var/log/** and **/var/log/containers/**.