

Brigded Broadband

- [RHEL](#)
 - [RHEL Repos](#)
- [Network](#)

RHEL

RHEL

RHEL Repos

Ensure Register

Check register Status with these commands. Output might not match exactly just general checks to see if anything looks weird.

```
sudo subscription-manager identity
```

```
system identity: 3634c02b-c3ce-4533-8151-8c7cc2d13318
name: RP-Host-03
org name: 7752782
org ID: 7752782
```

And

```
sudo subscription-manager status
```

```
+-----+
  System Status Details
+-----+
Overall Status: Registered
Content Access Mode is set to Simple Content Access. This host has access to content,
regardless of subscription status.
```

Register if not

```
sudo subscription-manager register --org=7752782 --activationkey=KEY_HERE
```

Repos

Repos should be added automatically after registering. Check to ensure the correct repos are enabled.

Ensure only these repos are enabled

- rhel-8-for-x86_64-baseos-rpms
- rhel-8-for-x86_64-appstream-rpms

```
sudo subscription-manager repos --list-enabled
```

Should show

```
+-----+
  Available Repositories in /etc/yum.repos.d/redhat.repo
+-----+
Repo ID:   rhel-8-for-x86_64-baseos-rpms
Repo Name: Red Hat Enterprise Linux 8 for x86_64 - BaseOS (RPMs)
Repo URL:  https://cdn.redhat.com/content/dist/rhel8/8.10/x86_64/baseos/os
Enabled:   1

Repo ID:   rhel-8-for-x86_64-appstream-rpms
Repo Name: Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Repo URL:  https://cdn.redhat.com/content/dist/rhel8/8.10/x86_64/appstream/os
Enabled:   1
```

Run updates

```
sudo dnf update -y
```

Failed EUS Repositories

Redhat adds Extended Update Support (EUS) repos to the list by default. These can be disabled as they are only required for unsupported dependencies.

Fix for most cases

```
sudo subscription-manager repos \  
  --disable=rhel-8-for-x86_64-baseos-eus-rpms \  
  --disable=rhel-8-for-x86_64-appstream-eus-rpms
```

```
sudo dnf update -y
```

Manual Checking

Check the listed repos

```
sudo subscription-manager repos --list-enabled
```

Disable repos by name

```
sudo subscription-manager repos --disable=rhel-8-for-x86_64-baseos-eus-rpms
```

Run upgrades

```
sudo dnf update -y
```

Network

Overview

There are 4 network interfaces.
Each interface bonds 2 physical network cards.

The 4 main vlans used are

- MDC RP Management
- NSP Client Network
- NSP Internal Network
- NSP Managed Network

Exact mappings and vlan information is in the spreadsheet:

[Network Interfaces and Vlans](#)

This sheet is for the RCPT hosts. The KCSW hosts are the exact same except the third IP field is +50.

| RCPT Vlan | |
|-----------|---------------|
| Vlan | 172.18.2.0/24 |
| Vlan Tag | 2102 |

| KCSW Vlan | |
|-----------|----------------|
| Vlan | 172.18.52.0/24 |
| Vlan Tag | 2152 |

Bond configurations

This creates the bond interfaces

```
nmcli connection add ipv4.method disabled ipv6.method ignore type bond mode active-backup
ifname bond1 con-name bond1
nmcli connection add type ethernet slave-type bond ifname ens6f0 master bond1 con-name bond1-
slave0
nmcli connection add type ethernet slave-type bond ifname ens6f1 master bond1 con-name bond1-
slave1

nmcli connection add ipv4.method disabled ipv6.method ignore type bond mode active-backup
ifname bond2 con-name bond2
nmcli connection add type ethernet slave-type bond ifname eno12409np1 master bond2 con-name
bond2-slave0
nmcli connection add type ethernet slave-type bond ifname eno12399np0 master bond2 con-name
bond2-slave1

nmcli connection add ipv4.method disabled ipv6.method ignore type bond mode active-backup
ifname bond3 con-name bond3
nmcli connection add type ethernet slave-type bond ifname eno12419np2 master bond3 con-name
bond3-slave0
nmcli connection add type ethernet slave-type bond ifname eno12429np3 master bond3 con-name
bond3-slave1

nmcli dev
```

Bridge configurations

This creates the bridges and adds them as slaves to the respective bond

```
nmcli con add type bridge ifname br1 con-name br1
nmcli con modify br1 ipv4.method disabled ipv6.method disabled
nmcli con modify bond1 master br1 slave-type bridge
nmcli con up br1

nmcli con add type bridge ifname br2 con-name br2
nmcli con modify br2 ipv4.method disabled ipv6.method disabled
nmcli con modify bond2 master br2 slave-type bridge
nmcli con up br2

nmcli con add type bridge ifname br3 con-name br3
nmcli con modify br3 ipv4.method disabled ipv6.method disabled
```

```
nmcli con modify bond3 master br3 slave-type bridge
nmcli con up br3

nmcli dev
```

VM ENV

This makes sure to use the system qemu instance instead of the userspace one when running virsh commands.

Run this before using any VM commands otherwise you will not find any VMs!

```
echo 'export LIBVIRT_DEFAULT_URI="qemu:///system"' >> ~/.bashrc
source ~/.bashrc
```

VM Nets

Use this script to add the bridges to the VMs

[add br vm.sh](#)

VM Bridge Assignments:

Client Bridge (br1) Vms:

- All Vms

NSP Internal Bridge (br2) Vms:

- All VMs excpet:
KCSW-WSNOC, KCSW-WSP, RCPT-WSNOC, RCPT-WSP

NSP Internal Bridge (br3) Vms:

- All VMs excpet:
KCSW-WSNOC, KCSW-WSP, RCPT-WSNOC, RCPT-WSP