

Please complete all quizzes, guided exercises, and labs for each chapter. Insert YES for each completed quiz and your screenshots in appropriate locations for each guided exercise/lab

Did you complete Quiz: Describe Networking Concepts

Guided Exercise: Validate Network Configuration

In this exercise, you inspect the network configuration of one of your servers.

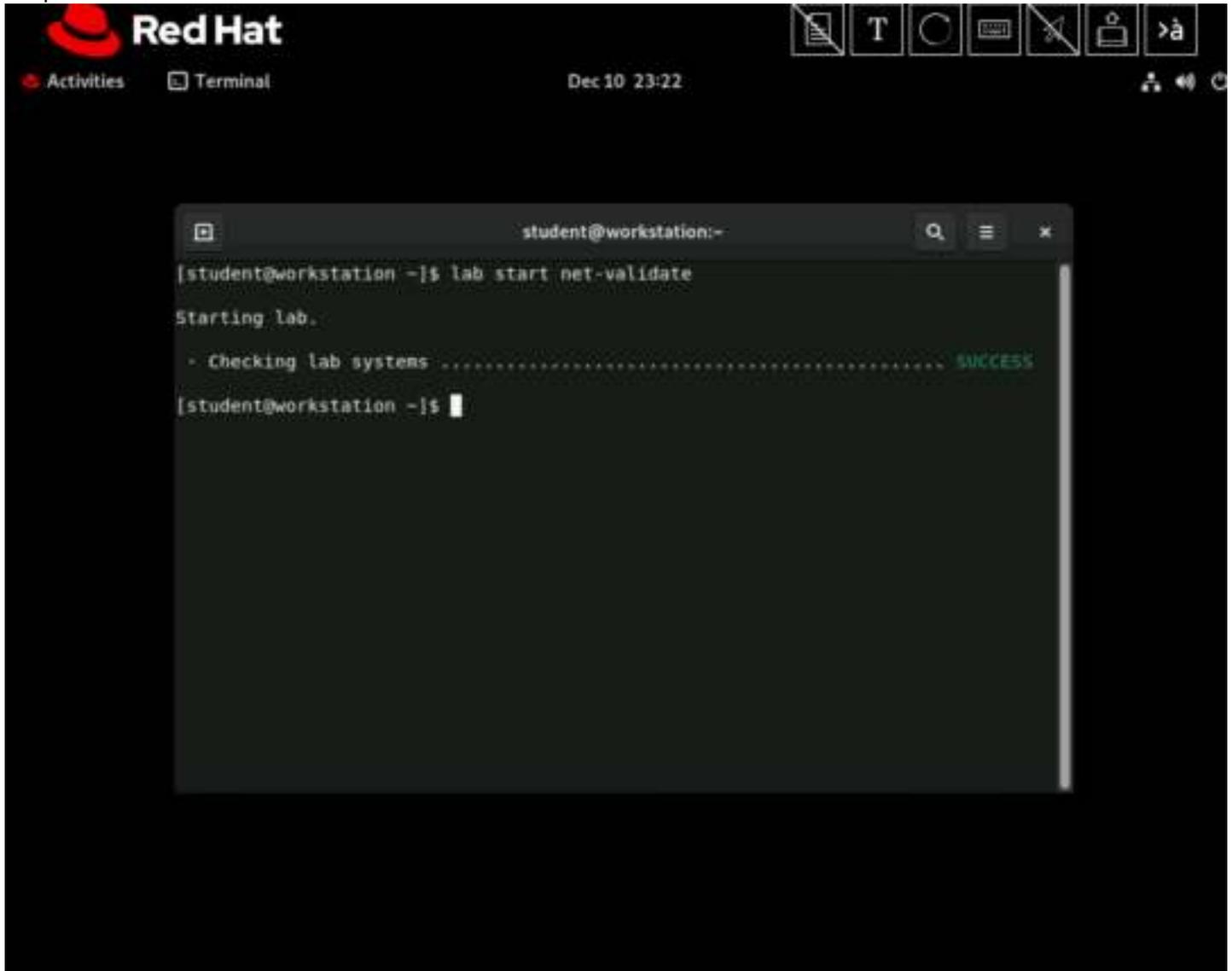
Outcomes

- Identify the current network interfaces and basic network addresses.

As the `student` user on the `workstation` machine, use the `lab` command to prepare your system for this exercise.

This command prepares your environment and ensures that all required resources are available.

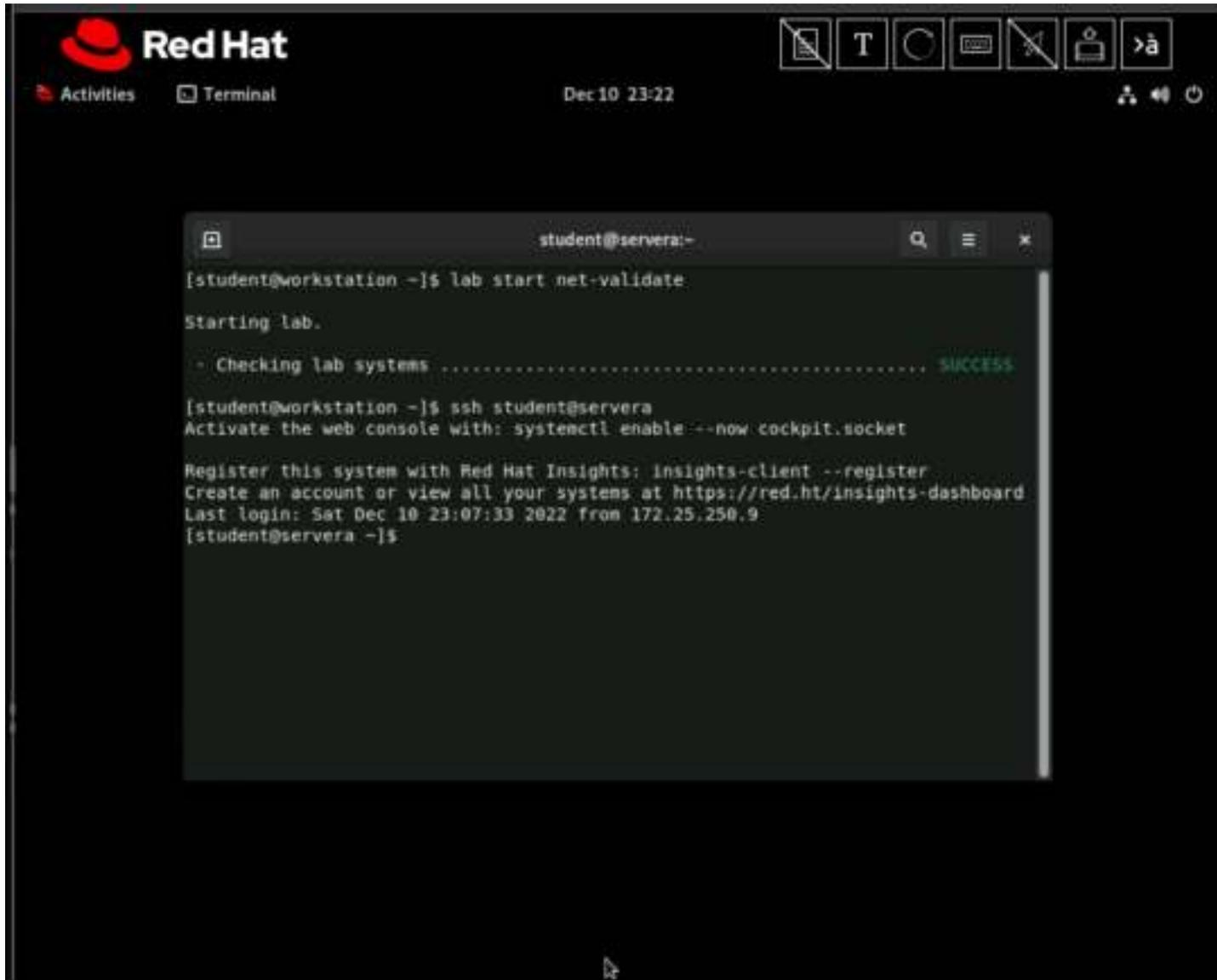
```
[student@workstation ~]$ lab start net-validate
```



Procedure 12.1. Instructions

1. Use the `ssh` command to log in to `servera` as the `student` user. The systems are configured to use SSH keys for authentication and passwordless access to `servera`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```



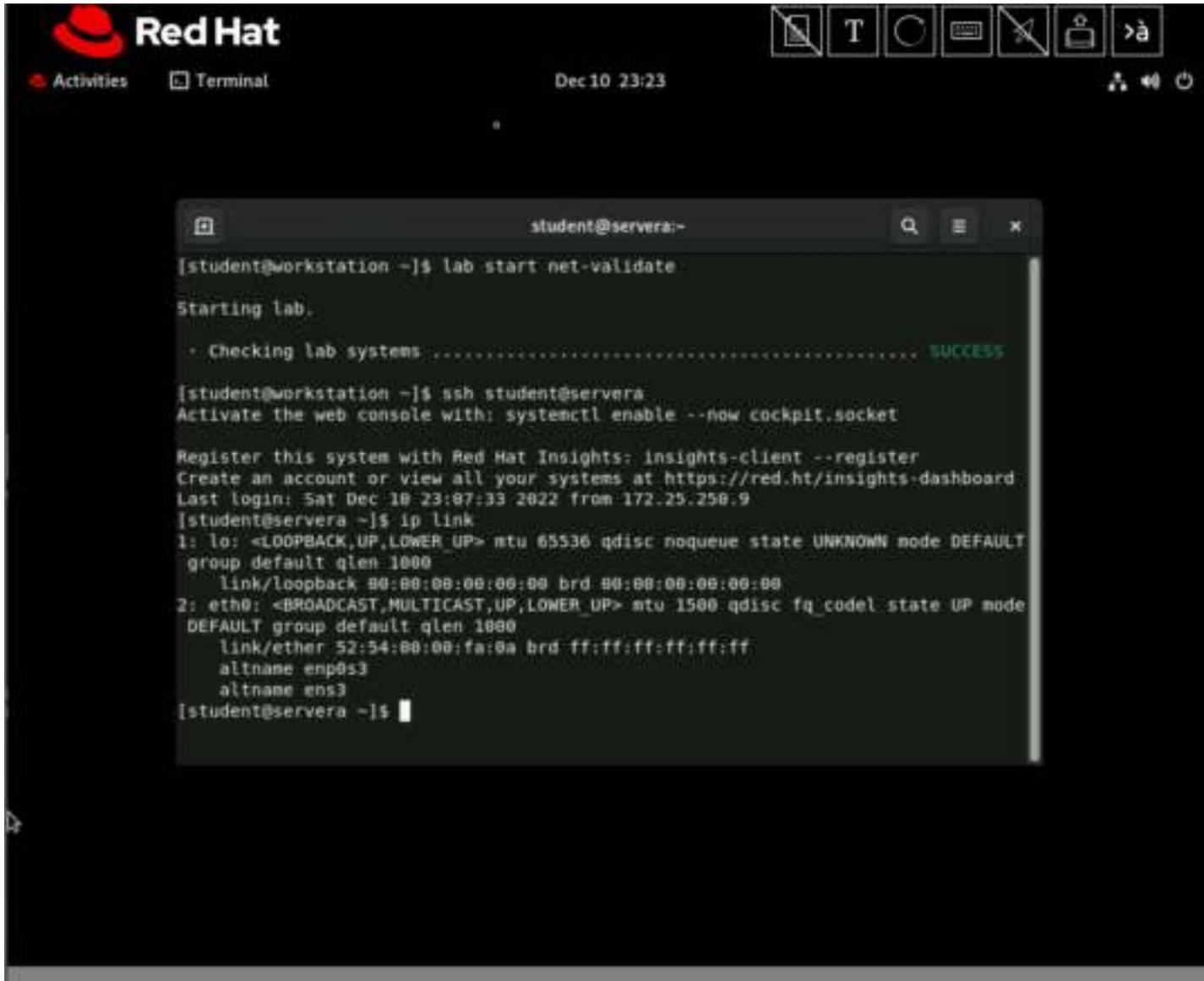
2. Locate the network interface name that is associated with the Ethernet address `52:54:00:00:fa:0a`. Record or remember this name and use it to replace the `enX` placeholder in subsequent commands.

Important

Network interface names are determined by their bus type and the detection order of devices during boot. Your network interface names will vary according to the course platform and hardware in use.

On your system, locate the interface name (such as `ens06` or `en1p2`) that is associated with the Ethernet address `52:54:00:00:fa:0a`. Use this interface name to replace the `enX` placeholder that is used throughout this exercise.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```



3. Display the current IP address and netmask for all interfaces.

```

[student@servera ~]$ ip -br addr
lo          UP          127.0.0.1/8 ::1/128
enX:       UP          172.25.250.10/24 fe80::3059:5462:198:58b2/64

```

4. Display the statistics for the enX interface.

```

[student@servera ~]$ ip -s link show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
   DEFAULT group default qlen 1000
   link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
   RX: bytes  packets  errors  dropped  overrun  mcast
   89014225  168251   0       154418   0         0
   TX: bytes  packets  errors  dropped  carrier  collsns
   608808    6090    0       0        0         0

```

The screenshot shows a terminal window titled 'student@servera:-' with the following output:

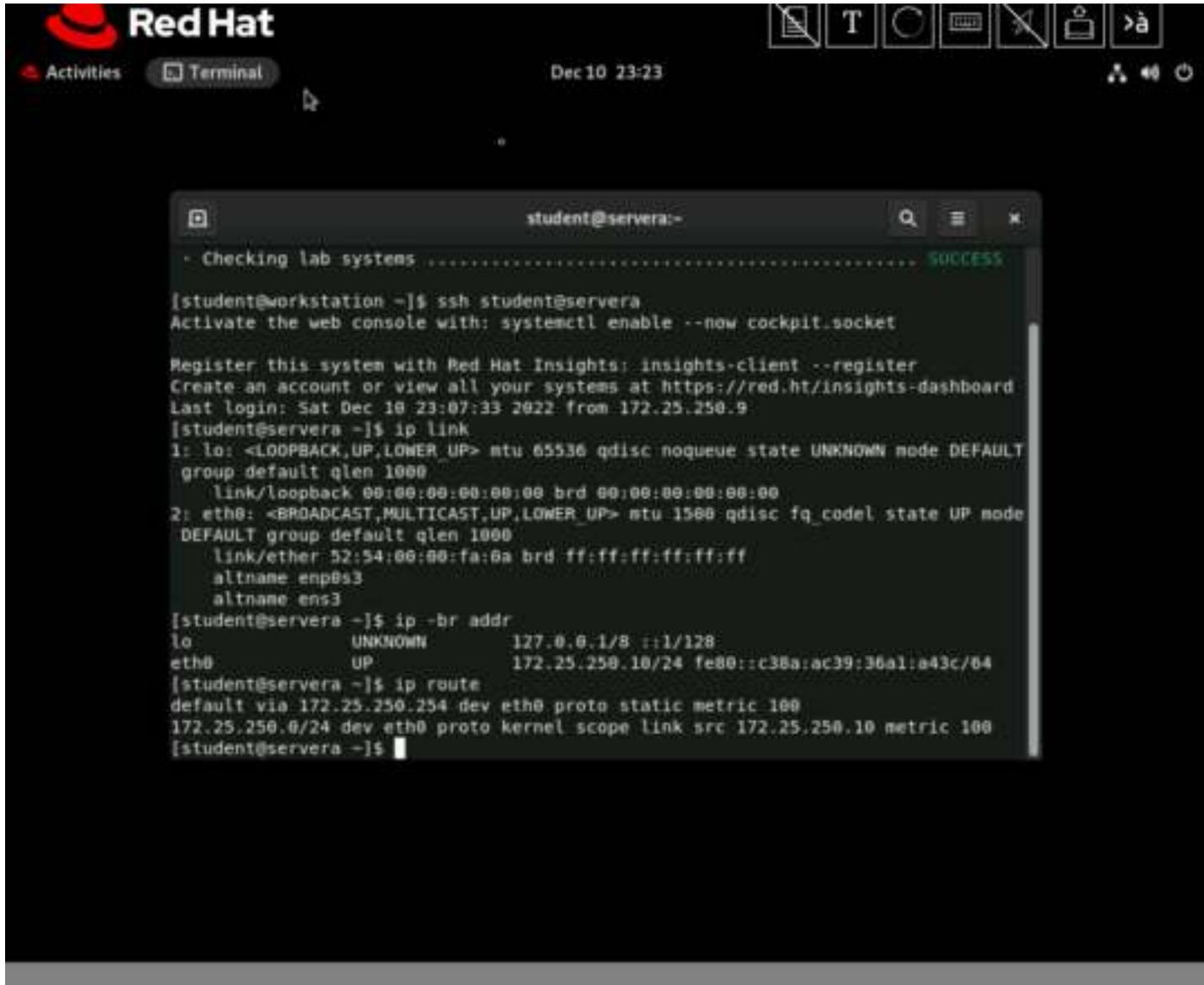
```
Starting lab.
- Checking lab systems ..... SUCCESS

[student@workstation ~]$ ssh student@servera
Activate the web console with: systemctl enable --now cockpit.socket

Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Sat Dec 18 23:07:33 2022 from 172.25.250.9
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
   group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 1500 qdisc fq_codel state UP mode
   DEFAULT group default qlen 1000
    link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
[student@servera ~]$ ip -br addr
lo          UNKNOWN      172.0.0.1/8  ::1/128
eth0       UP           172.25.250.10/24 fe80::c38a:ac39:36a1:a43c/64
[student@servera ~]$
```

5. Display the route information.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev enX proto static metric 100
172.25.250.0/24 dev enX proto kernel scope link src 172.25.250.10 metric 100
```



6. Verify that the router is accessible.

```

[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.196 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.436 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.361 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 49ms
rtt min/avg/max/mdev = 0.196/0.331/0.436/0.100 ms

```

The screenshot shows a terminal window on a Red Hat system. The terminal output is as follows:

```

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
   group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
   DEFAULT group default qlen 1000
   link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
   altname enp0s3
   altname ens3
[student@servera ~]$ ip -br addr
lo                UNKNOWN          127.0.0.1/8      ::1/128
eth0              UP              172.25.250.10/24 fe80::c38a:ac39:36a1:a43c/64
[student@servera ~]$ ip route
default via 172.25.250.254 dev eth0 proto static metric 100
172.25.250.0/24 dev eth0 proto kernel scope link src 172.25.250.10 metric 100
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data:
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.779 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.477 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.317 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.317/0.524/0.779/0.191 ms
[student@servera ~]$

```

7. Show all the hops between the local system and `classroom.example.com`.

```

[student@servera ~]$ tracert classroom.example.com
1?: [LOCALHOST] pmtu 1500
1: bastion.lab.example.com 0.337ms
1: bastion.lab.example.com 0.122ms
2: 172.25.254.254 0.602ms reached
Resume: pmtu 1500 hops 2 back 2

```

The screenshot shows a terminal window on a Red Hat system. The user is logged in as 'student' on a server named 'servera'. The terminal output shows the following commands and results:

```

altname enp0s3
altname ens3
[student@servera ~]$ ip -br addr
lo                UNKNOWN        127.0.0.1/8 ::1/128
eth0              UP              172.25.250.10/24 fe80::c38a:ac39:36a1:a43c/64
[student@servera ~]$ ip route
default via 172.25.250.254 dev eth0 proto static metric 100
172.25.250.0/24 dev eth0 proto kernel scope link src 172.25.250.10 metric 100
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data:
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.779 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.477 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.317 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.317/0.524/0.779/0.191 ms
[student@servera ~]$ tracepath classroom.example.com
  1?: [LOCALHOST]                pmtu 1500
  1:  bastion.lab.example.com      0.320ms
  1:  bastion.lab.example.com      0.114ms
  2:  172.25.254.254                0.371ms reached
Resume: pmtu 1500 hops 2 back 2
[student@servera ~]$

```

8. Display the listening TCP sockets on the local system.

```

[student@servera ~]$ ss -lt
State      Recv-Q  Send-Q      Local Address:Port      Peer Address:Port
LISTEN    0        128         0.0.0.0:sunrpc          0.0.0.0:*
LISTEN    0        128         0.0.0.0:ssh             0.0.0.0:*
LISTEN    0        128         [::]:sunrpc            [::]:*
LISTEN    0        128         [::]:ssh                [::]:*

```

The screenshot shows a terminal window on a Red Hat system. The terminal output includes network configuration for the default interface, followed by a ping test to 172.25.250.254, ping statistics, a tracepath to classroom.example.com, and a netstat command showing listening ports.

```

default via 172.25.250.254 dev eth0 proto static metric 100
172.25.250.0/24 dev eth0 proto kernel scope link src 172.25.250.10 metric 100
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data:
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.779 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.477 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.317 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.317/0.524/0.779/0.191 ms
[student@servera ~]$ tracepath classroom.example.com
 0: [LOCALHOST] pmtu 1500
 1: bastion.lab.example.com 0.320ms
 1: bastion.lab.example.com 0.114ms
 2: 172.25.254.254 0.371ms reached
Resume: pmtu 1500 hops 2 back 2
[student@servera ~]$ ss -lt
State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
LISTEN 0 4096 0.0.0.0:sunrpc 0.0.0.0:*
LISTEN 0 128 0.0.0.0:ssh 0.0.0.0:*
LISTEN 0 4096 :::sunrpc :::*
LISTEN 0 128 :::ssh :::*
[student@servera ~]$

```

- Return to the workstation system as the student user.

```

[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$

```

Finish

On the workstation machine, change to the student user home directory and use the lab command to complete this exercise. This step is important to ensure that resources from previous exercises do not impact upcoming exercises.

```
[student@workstation ~]$ lab finish net-validate
```



```
student@workstation:-
1: bastion.lab.example.com 0.114ms
2: 172.25.254.254 0.371ms reached
Resume: pmtu 1500 hops 2 back 2
[student@servera ~]$ ss -lt
State Recv-Q Send-Q Local Address:Port Peer Address:Port Process
LISTEN 0 4096 0.0.0.0:sunrpc 0.0.0.0:*
LISTEN 0 128 0.0.0.0:ssh 0.0.0.0:*
LISTEN 0 4096 [::]:sunrpc [::]:*
LISTEN 0 128 [::]:ssh [::]:*
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$ lab finish net-validate
Usage: lab [OPTIONS] COMMAND [ARGS]...
Try 'lab --help' for help.

Error: No such command 'finsh'.
[student@workstation ~]$ lab finish net-validate

Finishing lab.

- Checking lab systems ..... SUCCESS
[student@workstation ~]$
```

This concludes the section.

Guided Exercise: Configure Networking from the Command Line

In this exercise, you use the `nmcli` command to configure network settings.

Outcomes

- Update a network connection setting from DHCP to static.

As the `student` user on the `workstation` machine, use the `lab` command to prepare your system for this exercise.

This command prepares your environment and ensures that all required resources are available.

```
[student@workstation ~]$ lab start net-configure
```



Procedure 12.2. Instructions

1. Use the `ssh` command to log in to the `servera` machine as the `student` user.

```
[student@workstation ~]$ ssh student@servera
```

Chapter 12

```
...output omitted...  
[student@servera ~]$ sudo -i  
[sudo] password for student: student  
[root@servera ~]#
```



2. Display the network interface information.

Important

Network interface names are determined by their bus type and the detection order of devices during boot. Your network interface names might vary according to the course platform and hardware in use.

On your system, locate the interface name (such as `eth1`, `ens06`, or `enp0p2`) that is associated with the Ethernet address `52:54:00:00:fa:0a`. Use this interface name to replace the `eth0` placeholder throughout this exercise if different.

Locate the network interface name that is associated with the Ethernet address `52:54:00:00:fa:0a`. Record or remember this name and use it to replace the `eth0` placeholder in subsequent commands.

Chapter 12

```
[root@servera ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
DEFAULT group default qlen 1000
    link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
```



3. Use the `nmcli` command to view network settings.
 1. Use the `nmcli con show` to display all connections.

```
[root@servera ~]# nmcli con show
```

NAME	UUID	TYPE	DEVICE
System eth0	5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03	ethernet	eth0
System eth1	9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04	ethernet	--

2. Use the `nmcli con show --active` command to display only the active connections.

Your network interface name should appear under the `DEVICE` column of the output, and the name of the active connection for that device is listed under the `NAME` column. This exercise

assumes that the active connection is called `System eth0`. If the name of the active connection is different, then use that name instead of `System eth0` for the rest of this exercise.

```
[root@servera ~]# nmcli con show --active
NAME                UUID                                TYPE      DEVICE
System eth0        03da038a-3257-4722-a478-53055cc90128  ethernet  eth0
```

3. Display all configuration settings for the active connection.

```
[root@servera ~]# nmcli con show "System eth0"
connection.id:      System eth0
connection.uuid:    5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03
connection.stable-id: --
connection.type:    802-3-ethernet
connection.interface-name: eth0
connection.autoconnect: yes
...output omitted...
ipv4.method:        manual
ipv4.dns:            172.25.250.254,2.2.2.2
ipv4.dns-search:    lab.example.com,example.com
ipv4.dns-options:   --
ipv4.dns-priority:  0
ipv4.addresses:     172.25.250.10/24
ipv4.gateway:       172.25.250.254
...output omitted...
ipv6.method:        ignore
ipv6.dns:           --
ipv6.dns-search:    --
ipv6.dns-options:   --
ipv6.dns-priority:  0
ipv6.addresses:     --
ipv6.gateway:       --
ipv6.routes:        --
...output omitted...
GENERAL.NAME:       System eth0
GENERAL.UUID:       5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03
GENERAL.DEVICES:    eth0
GENERAL.IP-IFACE:   eth0
GENERAL.STATE:      activated
GENERAL.DEFAULT:    yes
```

4. Show the device status.

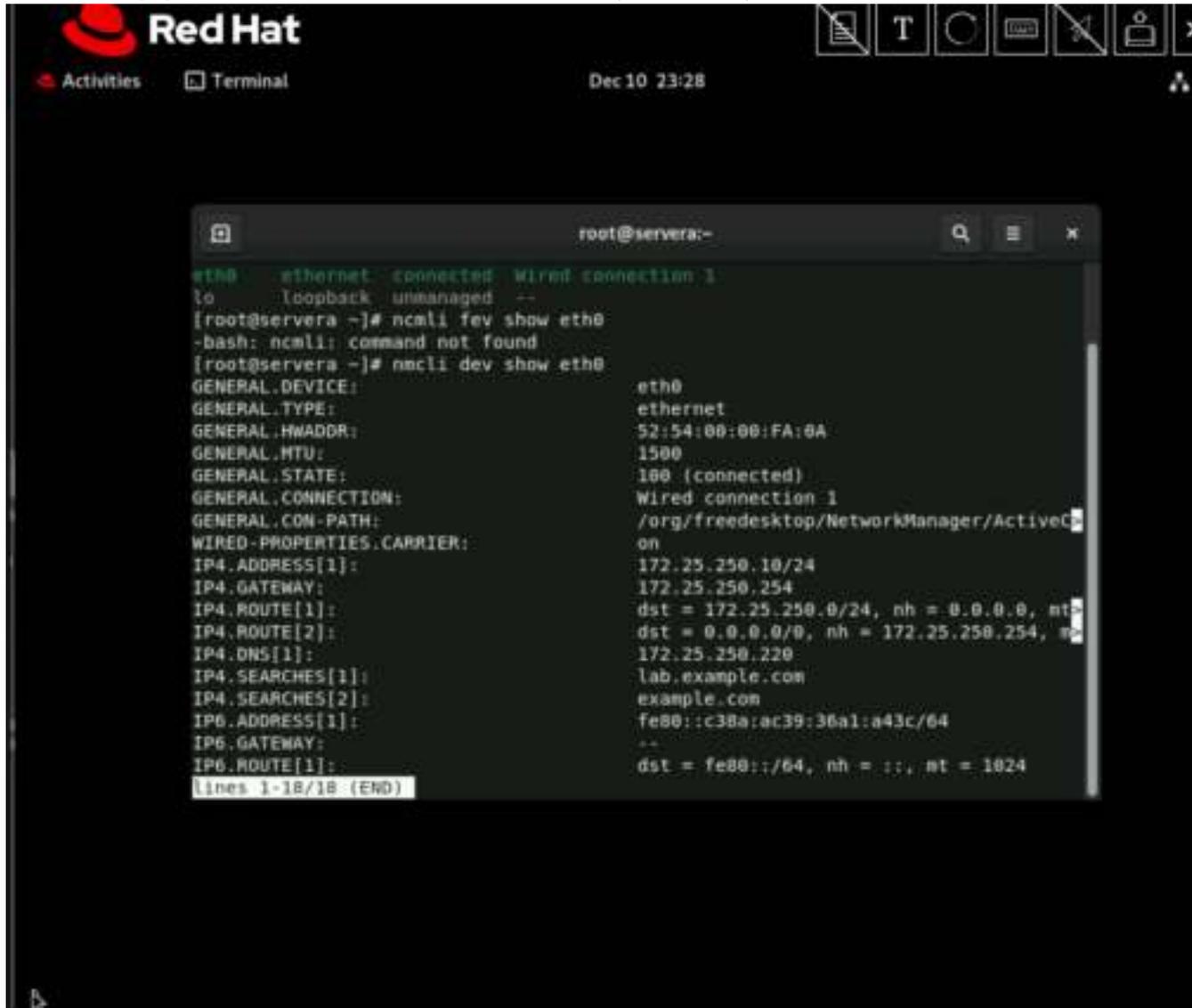
```
[root@servera ~]# nmcli dev status
DEVICE  TYPE      STATE      CONNECTION
eth0    ethernet  connected  System eth0
lo      loopback  unmanaged  --
```

5. Display the settings for the `eth0` device.

```
[root@servera ~]# nmcli dev show eth0
GENERAL.DEVICE:     eth0
GENERAL.TYPE:       ethernet
GENERAL.HWADDR:     52:54:00:00:FA:0A
GENERAL.MTU:        1500
GENERAL.STATE:      100 (connected)
GENERAL.CONNECTION: System eth0
GENERAL.CON-PATH:   /org/freedesktop/NetworkManager/ActiveConnection/3
```

Chapter 12

```
WIRED-PROPERTIES.CARRIER: on
IP4.ADDRESS[1]: 172.25.250.10/24
IP4.GATEWAY: 172.25.250.254
IP4.ROUTE[1]: dst = 172.25.250.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]: dst = 0.0.0.0/0, nh = 172.25.250.254, mt = 100
IP4.DNS[1]: 172.25.250.254
IP4.SEARCHES[1]: lab.example.com
IP4.SEARCHES[2]: example.com
IP6.ADDRESS[1]: fe80::5054:ff:fe00:fa0a/64
IP6.GATEWAY: --
IP6.ROUTE[1]: dst = fe80::/64, nh = ::, mt = 256
```



The screenshot shows a terminal window titled "root@servera:-" with the following output:

```
eth0 ethernet connected wired connection 1
lo loopback unmanaged --
[root@servera ~]# nmcli fev show eth0
-bash: nmcli: command not found
[root@servera ~]# nmcli dev show eth0
GENERAL.DEVICE: eth0
GENERAL.TYPE: ethernet
GENERAL.HWADDR: 52:54:00:08:FA:0A
GENERAL.MTU: 1500
GENERAL.STATE: 100 (connected)
GENERAL.CONNECTION: Wired connection 1
GENERAL.CON-PATH: /org/freedesktop/NetworkManager/ActiveC
WIRED-PROPERTIES.CARRIER: on
IP4.ADDRESS[1]: 172.25.250.10/24
IP4.GATEWAY: 172.25.250.254
IP4.ROUTE[1]: dst = 172.25.250.0/24, nh = 0.0.0.0, mt
IP4.ROUTE[2]: dst = 0.0.0.0/0, nh = 172.25.250.254, m
IP4.DNS[1]: 172.25.250.220
IP4.SEARCHES[1]: lab.example.com
IP4.SEARCHES[2]: example.com
IP6.ADDRESS[1]: fe80::c38a:ac39:36a1:a43c/64
IP6.GATEWAY: --
IP6.ROUTE[1]: dst = fe80::/64, nh = ::, mt = 1024
lines 1-18/18 (END)
```

4. Create a static connection with the same IPv4 address, network prefix, and default gateway as the active connection. Name the new connection `static-addr`.

Warning

Because access to your machine is provided over the primary network connection, setting incorrect values during network configuration might make your machine unreachable. If your machine is unreachable, then use the Reset button above what used to be your machine's graphical display and try again.

Chapter 12

```
[root@servera ~]# nmcli con add con-name static-addr \  
ifname eth0 type ethernet ipv4.method manual \  
ipv4.addresses 172.25.250.10/24 ipv4.gateway 172.25.250.254  
Connection 'static-addr' (c242697d-498e-481c-b974-5ae11d2a0291) successfully added.
```



5. Modify the new connection to add the DNS setting.

```
[root@servera ~]# nmcli con mod static-addr ipv4.dns 172.25.250.254
```



6. Display and activate the new connection.

1. View all connections.

```

[root@servera ~]# nmcli con show
NAME                UUID                                  TYPE      DEVICE
System eth0        5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03  ethernet  eth0
static-addr        e4cf52d3-40fc-41b3-b5e8-cf280157f3bb  ethernet  --
System eth1        9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04  ethernet  --

```

2. View the active connections.

```

[root@servera ~]# nmcli con show --active
NAME                UUID                                  TYPE      DEVICE
System eth0        5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03  ethernet  eth0

```

3. Activate the new static-addr connection.

```

[root@servera ~]# nmcli con up static-addr
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/4)

```

4. Verify the new active connection.

```
[root@servera ~]# nmcli con show --active
```

NAME	UUID	TYPE	DEVICE
static-addr	e4cf52d3-40fc-41b3-b5e8-cf280157f3bb	ethernet	eth0

```

root@servera:~
IP4.SEARCHES[1]: lab.example.com
IP4.SEARCHES[2]: example.com
IP6.ADDRESS[1]: fe80::c38a:ac39:36a1:a43c/64
IP6.GATEWAY: --
IP6.ROUTE[1]: dst = fe80::/64, nh = ::, mt = 1024
[root@servera ~]# nmcli con add con-name static-addr ifname eth0 type ethernet i
pv4.method manual ipv4.addresses 172.25.250.18/24 ipv4.gateway 172.25.250.254
Connection 'static-addr' (57bd14ee-cebc-46c4-9bd0-b367bd0919c4) successfully add
ed.
[root@servera ~]# nmcli con mod static-addr ipv4.dns 172.25.250.254
[root@servera ~]# nmcli con show
NAME                UUID                                TYPE      DEVICE
Wired connection 1  ec3a15fb-2e28-3254-9433-98c60981e924 ethernet eth0
static-addr         57bd14ee-cebc-46c4-9bd0-b367bd0919c4 ethernet --
[root@servera ~]# nmcli con show --active
NAME                UUID                                TYPE      DEVICE
Wired connection 1  ec3a15fb-2e28-3254-9433-98c60981e924 ethernet eth0
[root@servera ~]# nmcli con up static-addr
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkMa
nager/ActiveConnection/2)
[root@servera ~]# nmcli con show --active
NAME                UUID                                TYPE      DEVICE
static-addr         57bd14ee-cebc-46c4-9bd0-b367bd0919c4 ethernet eth0
[root@servera ~]#

```

7. Update the previous connection so that it does not start at boot. Verify that the `static-addr` connection is used when the system reboots.
 1. Disable the original connection so that it does not start automatically at boot.

```
[root@servera ~]# nmcli con mod "System eth0" \
connection.autoconnect no
```

2. Reboot the system.

```
[root@servera ~]# systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Chapter 12

3. Log in to the `servera` machine and verify that the `static-addr` connection is the active connection.

```
[student@workstation ~]$ ssh student@servera
[student@servera ~]$ nmcli con show --active
NAME                UUID                                TYPE      DEVICE
static-addr         e4cf52d3-40fc-41b3-b5e8-cf280157f3bb  ethernet  eth0
```



8. Test connectivity by using the new network addresses.
 1. Verify the IP address.

```
[student@servera ~]$ ip -br addr show eth0
eth0      UP          172.25.250.10/24 fe80::47cd:2076:4a6b:e730/64
```

2. Verify the default gateway.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev eth0 proto static metric 100
172.25.250.0/24 dev eth0 proto kernel scope link src 172.25.250.10 metric 100
```

3. Ping the DNS address.

Chapter 12

```
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.669 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.294 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.283 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.283/0.415/0.669/0.179 ms
```

4. Return to the workstation system as the student user.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```



Finish

Chapter 12

On the workstation machine, change to the `student` user home directory and use the `lab` command to complete this exercise. This step is important to ensure that resources from previous exercises do not impact upcoming exercises.

```
[student@workstation ~]$ lab finish net-configure
```

This concludes the section.



```
student@workstation:-
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.261 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.288 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.261/0.384/0.605/0.156 ms
[root@servera ~]# exit
logout
[student@servera ~]$ lab finish net-configure
-bash: lab: command not found
[student@servera ~]$ exit
logout
connection to servera closed.
[student@workstation ~]$ lab finish net-configure

Finishing lab.

- Checking lab systems ..... SUCCESS
- Activating default nmcli profile ..... SUCCESS
- Auto-connecting to default profile ..... SUCCESS
- Removing static connection profile ..... SUCCESS
- Removing backup nmcli profile ..... SUCCESS

[student@workstation ~]$
```

Guided Exercise: Edit Network Configuration Files

In this exercise, you manually modify network configuration files and ensure that the new settings take effect.

Outcomes

- Configure additional network addresses on each system.

As the `student` user on the `workstation` machine, use the `lab` command to prepare your system for this exercise.

This command prepares your environment and ensures that all required resources are available.

```
[student@workstation ~]$ lab start net-edit
```

Procedure 12.3. Instructions

1. On the `workstation` machine, use the `ssh` command to log in to the `servera` machine as the `student` user.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```



2. Locate network interface names with the `ip link` command.

Important

Network interface names are determined by their bus type and the detection order of devices during boot. Your network interface names might vary according to the course platform and hardware in use.

Locate the network interface name that is associated with the Ethernet address on your system. Record or remember this name and use it to replace the `enX` placeholder in subsequent commands. The active connection is called `Wired connection 1` and the configuration is in the `/etc/NetworkManager/system-connections/"Wired connection 1.nmconnection"` file.

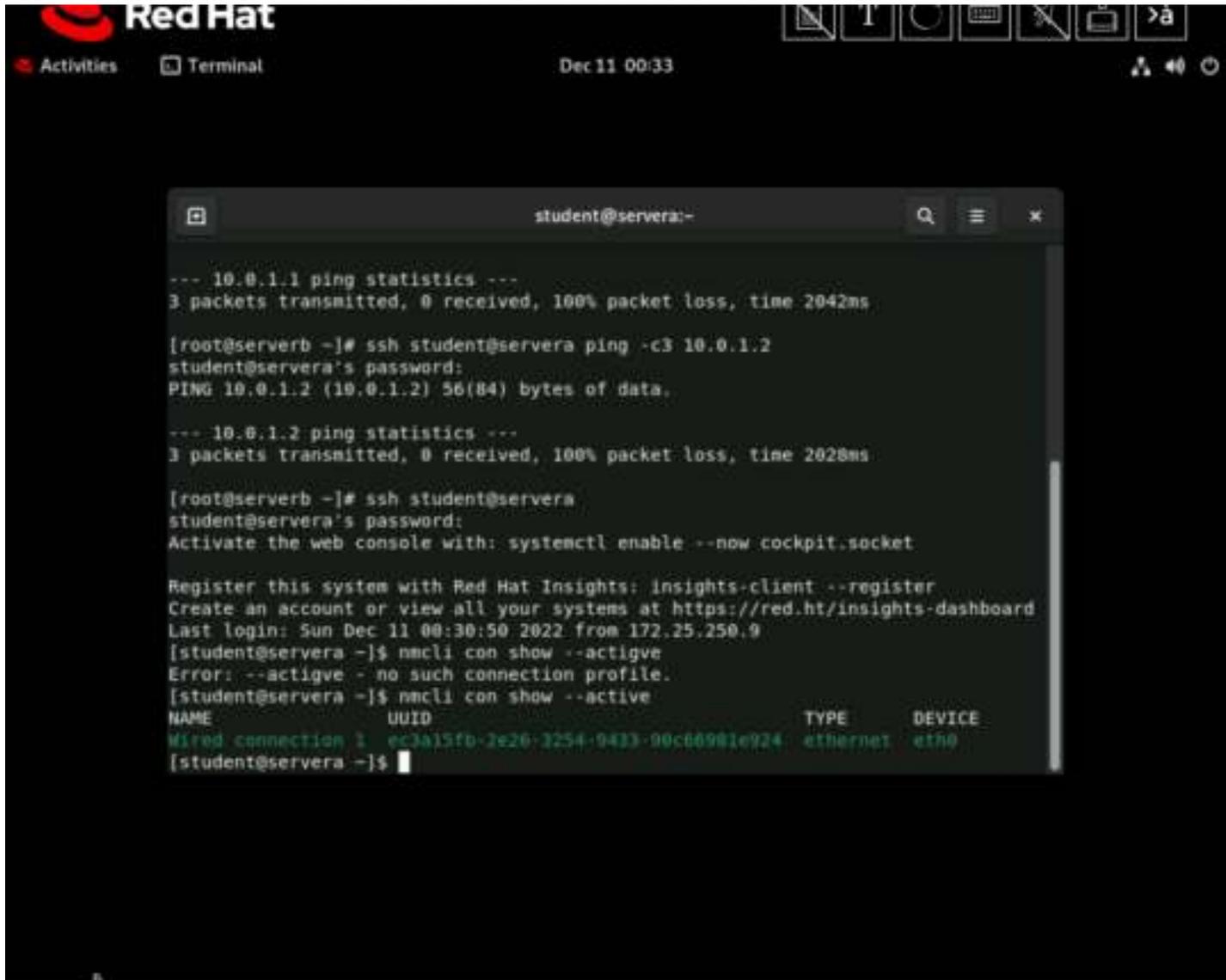
```

[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
   group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
   DEFAULT group default qlen 1000
   link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff

```

Chapter 12

```
altname enp0s3
altname ens3
[student@servera ~]$ nmcli con show --active
NAME                UUID                                     TYPE      DEVICE
Wired connection 1  a98933fa-25c0-36a2-b3cd-c056f41758fe  ethernet  eth0
```



```
[student@servera ~]$ ls /etc/NetworkManager/system-connections/
'Wired connection 1.nmconnection'
```

```

3 packets transmitted, 0 received, 100% packet loss, time 2042ms

[root@serverb ~]# ssh student@servera ping -c3 10.0.1.2
student@servera's password:
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data:

--- 10.0.1.2 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2028ms

[root@serverb ~]# ssh student@servera
student@servera's password:
Activate the web console with: systemctl enable --now cockpit.socket

Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Sun Dec 11 00:38:50 2022 from 172.25.250.9
[student@servera ~]$ nmcli con show --actigve
Error: --actigve - no such connection profile.
[student@servera ~]$ nmcli con show --active
NAME                UUID                                  TYPE      DEVICE
Wired connection 1  ec3a15fb-2e26-3254-9433-96c86981e934 ethernet eth0
[student@servera ~]$ ls /etc/NetworkManager/system-connections/
'Wired connection 1.nmconnection'
[student@servera ~]$

```

3. On the servera machine, switch to the root user, and then edit the `/etc/NetworkManager/system-connections/"Wired connection 1.nmconnection"` file to add the `10.0.1.1/24` address.
 1. Use the `sudo -i` command to switch to the root user.

```

[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#

```

2. Edit the configuration file. Add the `10.0.1.1/24` address as the second address below the first address in the file.

```

[root@servera ~]# vim /etc/NetworkManager/system-connections/"Wired
connection 1.nmconnection"
..output omitted...
[ipv4]
address1=172.25.250.10/24,172.25.250.254
address2=10.0.1.1/24
...output omitted...

```

Chapter 12

4. Activate the new network address with the `nmcli` command.
 1. Reload the configuration changes for NetworkManager to read the changes.

```
[root@servera ~]# nmcli con reload
```

2. Activate the connection with the changes.

```
[root@servera ~]# nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/2)
```

5. Verify that the new IP address is assigned successfully.

```
[root@servera ~]# ip -br addr show enX
eth0:          UP          172.25.250.10/24 10.0.1.1/24 fe80::6fed:5a11:4ad4:1bcf/64
```

6. Return to the workstation machine as the student user.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

7. On the serverb machine, edit the `/etc/NetworkManager/system-connections/"Wired connection 1.nmconnection"` file to add an address of `10.0.1.2/24` and load the new configuration.
 1. Log in to the servera machine as the student user and switch to the root user.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$ sudo -i
[sudo] password for student: student
[root@serverb ~]#
```



2. Edit the configuration file. Add the 10.0.1.2/24 address as the second address below the first address in the file.

```

[root@serverb ~]# vim /etc/NetworkManager/system-connections/"Wired
connection 1.nmconnection"
address1=172.25.250.11/24,172.25.250.254
address2=10.0.1.2/24

```

3. Reload the configuration changes for NetworkManager to read the changes.

```

[root@serverb ~]# nmcli con reload

```

4. Activate the connection with the changes.

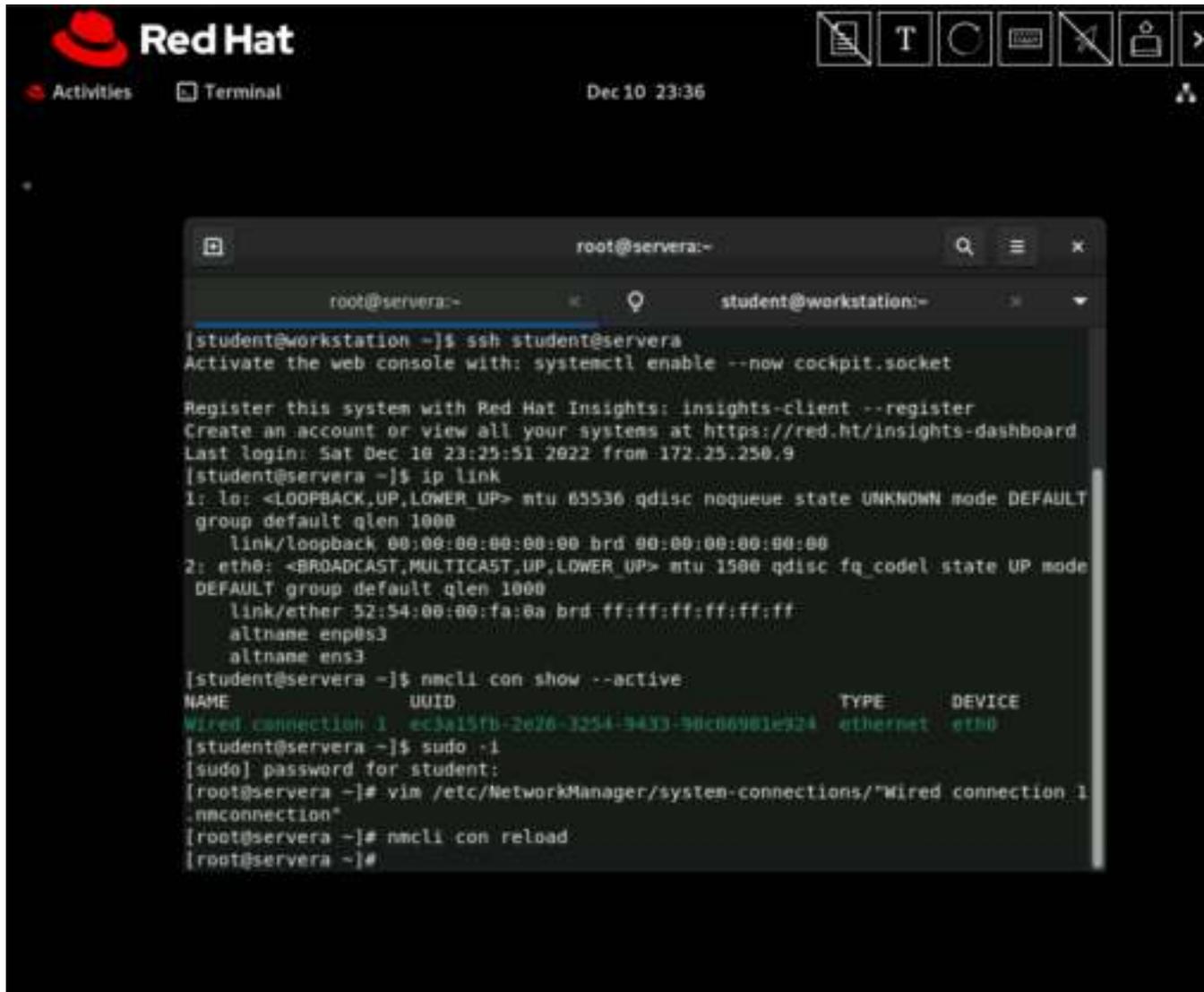
```

[root@serverb ~]# nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/2)

```

5. Verify that the new IP address is assigned successfully.

```
[root@serverb ~]# ip -br addr show enX
eth0      UP      172.25.250.11/24 10.0.1.2/24 fe80::6be8:6651:4280:892c/64
```



8. Test connectivity between the `servera` and `serverb` machines by using the new network addresses.
 1. From the `serverb` machine, ping the new address of the `servera` machine.

```
[root@serverb ~]# ping -c3 10.0.1.1
PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data:
64 bytes from 10.0.1.1: icmp_seq=1 ttl=64 time=1.30 ms
64 bytes from 10.0.1.1: icmp_seq=2 ttl=64 time=0.983 ms
64 bytes from 10.0.1.1: icmp_seq=3 ttl=64 time=0.312 ms

--- 10.0.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.312/0.864/1.297/0.410 ms
```

2. Return to the workstation machine as the student user.

```
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
```

Chapter 12

```
[student@workstation ~]$
```

3. Access the `servera` machine as the `student` user to ping the new address of the `serverb` machine.

```
[student@workstation ~]$ ssh student@servera ping -c3 10.0.1.2
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data.
64 bytes from 10.0.1.2: icmp_seq=1 ttl=64 time=0.876 ms
64 bytes from 10.0.1.2: icmp_seq=2 ttl=64 time=0.310 ms
64 bytes from 10.0.1.2: icmp_seq=3 ttl=64 time=0.289 ms

--- 10.0.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2047ms
rtt min/avg/max/mdev = 0.289/0.491/0.876/0.271 ms
```

Finish

On the `workstation` machine, change to the `student` user home directory and use the `lab` command to complete this exercise. This step is important to ensure that resources from previous exercises do not impact upcoming exercises.

```
[student@workstation ~]$ lab finish net-edit
```

This concludes the section.



```
nmconnection*
[root@serverb ~]# sudo vim /etc/NetworkManager/system-connections/~Wired connection
on 1.nmconnection*
[root@serverb ~]# lab finish net-edit
-bash: lab: command not found
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@servera ~]$ exit
logout
Connection to servera closed.
[root@serverb ~]# exit
logout
Connection to serverb closed.
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$ lab finish net-edit

Finishing lab.
· Checking lab systems ..... SUCCESS
```

Guided Exercise: Configure Hostnames and Name Resolution

In this exercise, you manually configure the system's static hostname, `/etc/hosts` file, and DNS name resolver.

Outcomes

- Set a customized hostname.
- Configure name resolution settings.

As the `student` user on the `workstation` machine, use the `lab` command to prepare your system for this exercise.

This command ensures that all required resources are available.

```
[student@workstation ~]$ lab start net-hostnames
```

Procedure 12.4. Instructions

1. Log in to `servera` as the `student` user and switch to `root` user.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```



2. View the current hostname settings.
 1. Display the current hostname.

```
[root@servera ~]# hostname
servera.lab.example.com
```

```

root@servera:~#
- Checking lab systems ..... SUCCESS
- Restoring hostname file ..... SUCCESS
- Restoring hosts file ..... SUCCESS

[student@workstation ~]$ lab start net-hostnames

Starting lab.

- Checking lab systems ..... SUCCESS
- Backing up hostname file ..... SUCCESS
- Backing up hosts file ..... SUCCESS

[student@workstation ~]$ ssh student@servera
Activate the web console with: systemctl enable --now cockpit.socket

Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Sun Dec 11 00:33:00 2022 from 172.25.250.11
[student@servera ~]$ sudo -i
[sudo] password for student:
[root@servera ~]# hostname
servera.lab.example.com
[root@servera ~]#

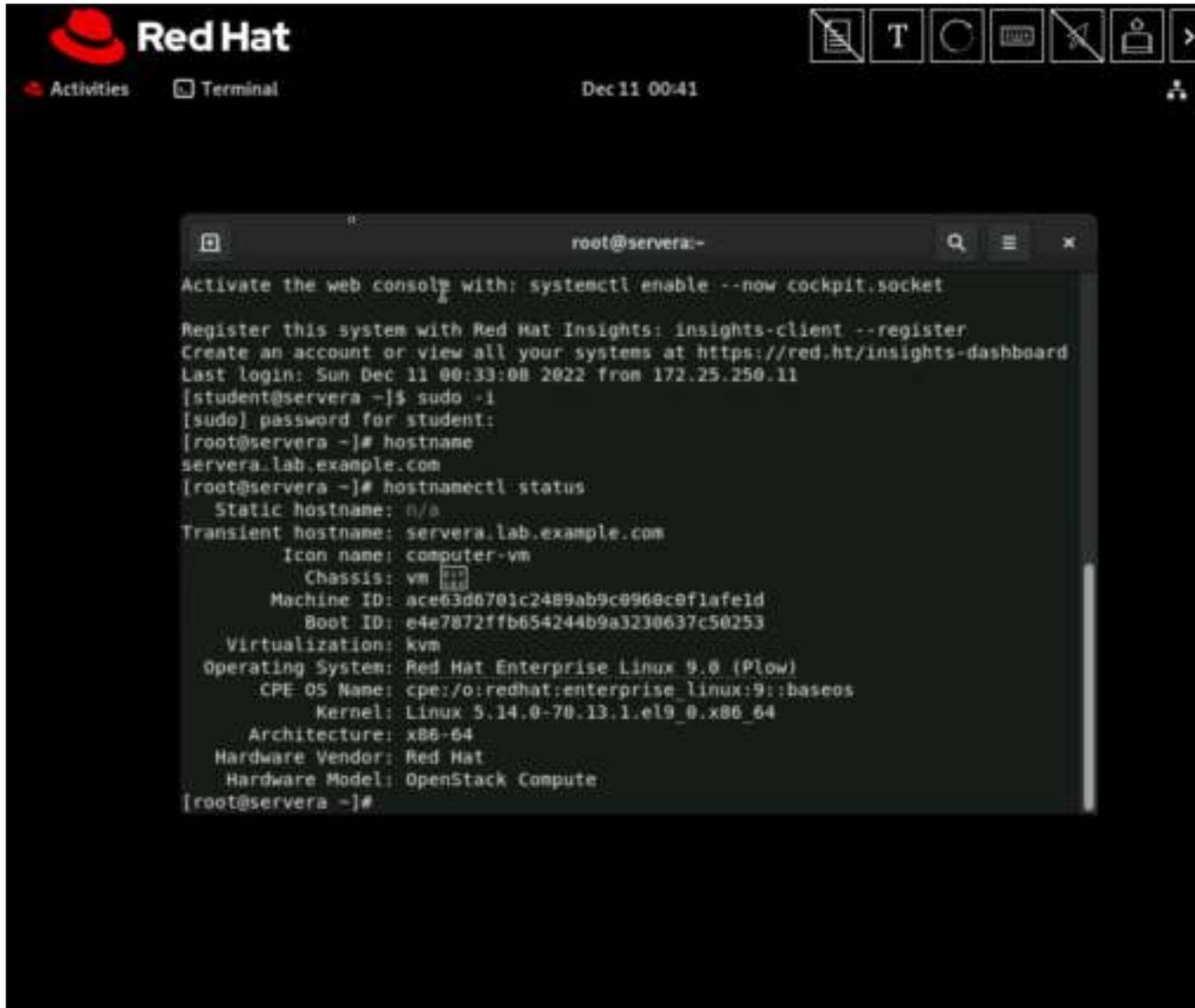
```

2. Display the hostname status. Note the transient hostname that is obtained from DHCP or mDNS.

```

[root@servera ~]# hostnamectl status
  Static hostname: n/a
Transient hostname: servera.lab.example.com
    Icon name: computer-vm
     Chassis: vm
  Machine ID: 63b272eae8d5443ca7aaa5593479b25f
   Boot ID: ef299e0e957041ee81d0617fc98ce5ef
Virtualization: kvm
Operating System: Red Hat Enterprise Linux 9.0 (Plow)
  CPE OS Name: cpe:/o:redhat:enterprise_linux:9::baseos
    Kernel: Linux 5.14.0-70.el9.x86_64
  Architecture: x86-64
Hardware Vendor: Red Hat
Hardware Model: OpenStack Compute

```



3. Set a static hostname to match the current transient hostname.
 1. Change the hostname and the hostname configuration file.

```
[root@servera ~]# hostnamectl set-hostname \
servera.lab.example.com
```

2. View the content of the `/etc/hostname` file, which provides the hostname at network start.

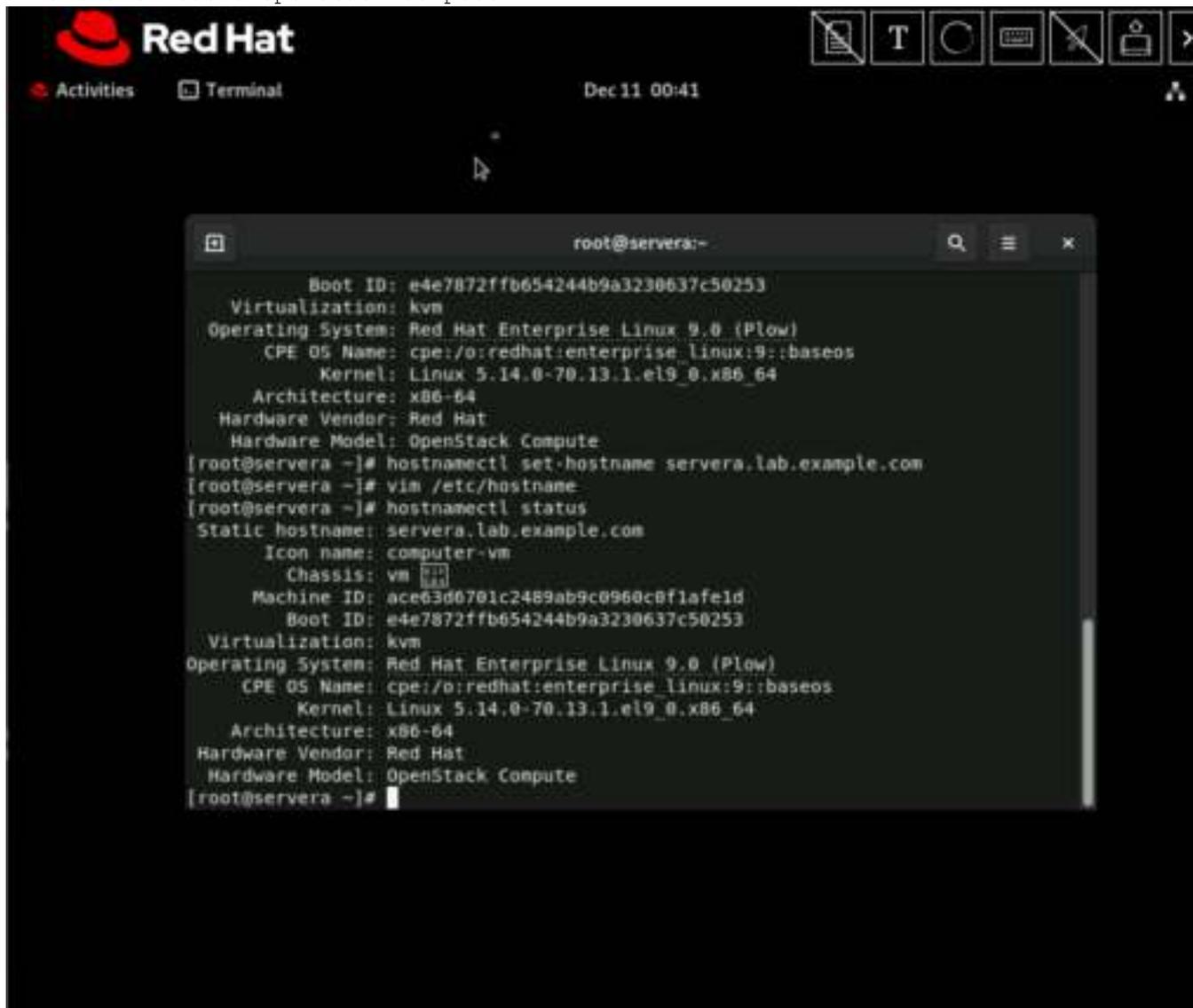
```
servera.lab.example.com
```

3. Display the hostname status. The transient hostname is not shown, now that a static hostname is configured.

```
[root@servera ~]# hostnamectl status
Static hostname: servera.lab.example.com
Icon name: computer-vm
Chassis: vm
Machine ID: 63b272eae8d5443ca7aaa5593479b25f
Boot ID: ef299e0e957041ee81d0617fc98ce5ef
Virtualization: kvm
```

Chapter 12

```
Operating System: Red Hat Enterprise Linux 9.0 (Plow)
  CPE OS Name: cpe:/o:redhat:enterprise_linux:9::baseos
    Kernel: Linux 5.14.0-70.el9.x86_64
  Architecture: x86-64
Hardware Vendor: Red Hat
Hardware Model: OpenStack Compute
```



4. Temporarily change the hostname to testname.

1. Change the hostname.

```
[root@servera ~]# hostname testname
```

2. Display the current hostname.

```
[root@servera ~]# hostname
testname
```

3. View the content of the `/etc/hostname` file, which provides the hostname at network start.

```
servera.lab.example.com
```

4. Reboot the system.

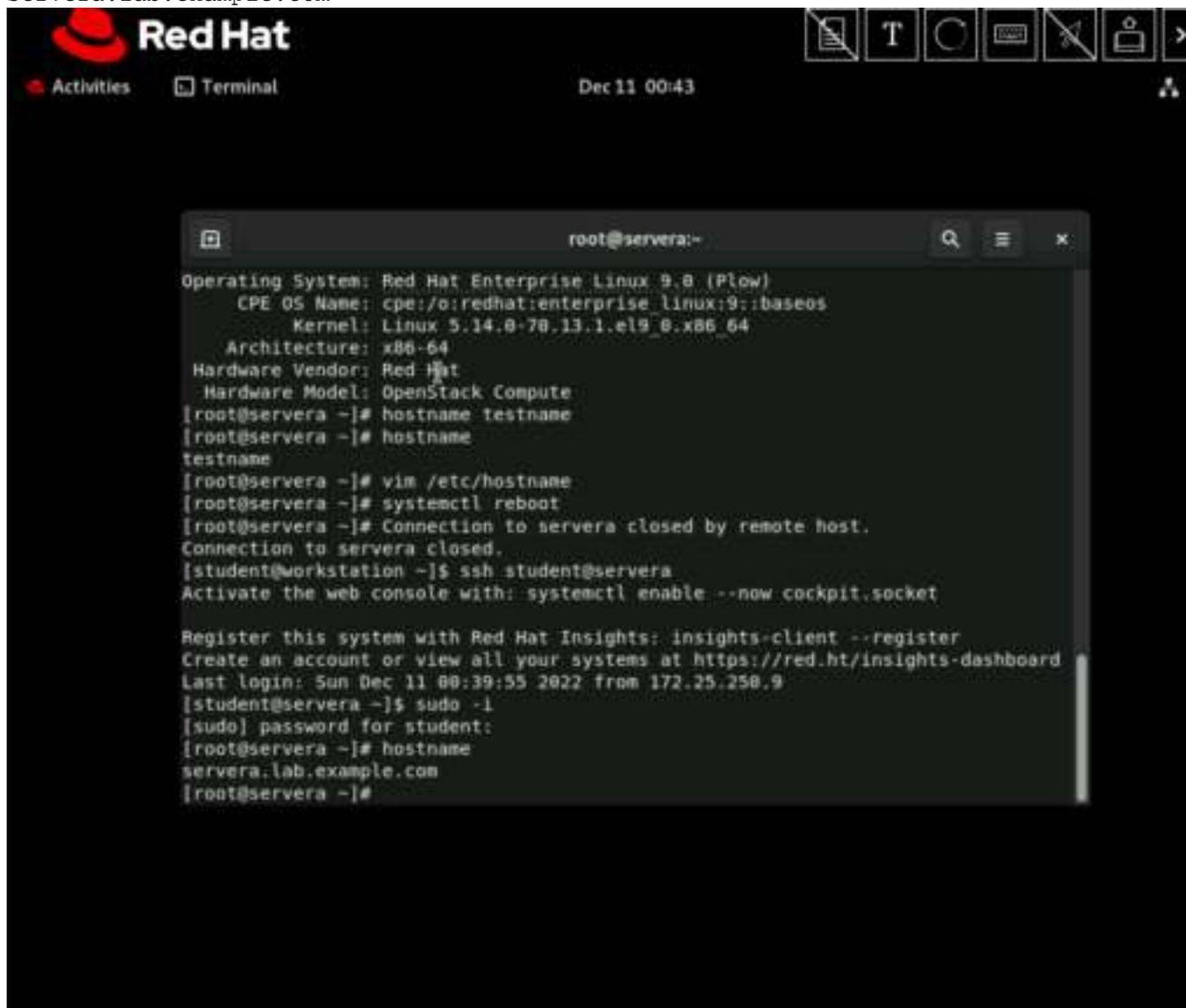
```
[root@servera ~]# systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

5. Log in to servera as the student user and switch to root user.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

6. Display the current hostname.

```
[root@servera ~]# hostname
servera.lab.example.com
```



Chapter 12

5. Add `class` as a local nickname for the classroom server, and ensure that you can ping the server with that nickname.
 1. Look up the IP address of `classroom.example.com`.

```
[root@servera ~]# host classroom.example.com  
classroom.example.com has address 172.25.254.254
```

2. Update the `/etc/hosts` file to add `class` to access the IP address `172.25.254.254`. The following example shows the expected content of the `/etc/hosts` file.

```
[root@servera ~]# vim /etc/hosts  
127.0.0.1    localhost localhost.localdomain localhost4  
localhost4.localhostdomain4  
::1         localhost localhost.localdomain localhost6  
localhost6.localhostdomain6  
172.25.254.254 classroom.example.com classroom class
```

3. Look up the IP address of `class`.

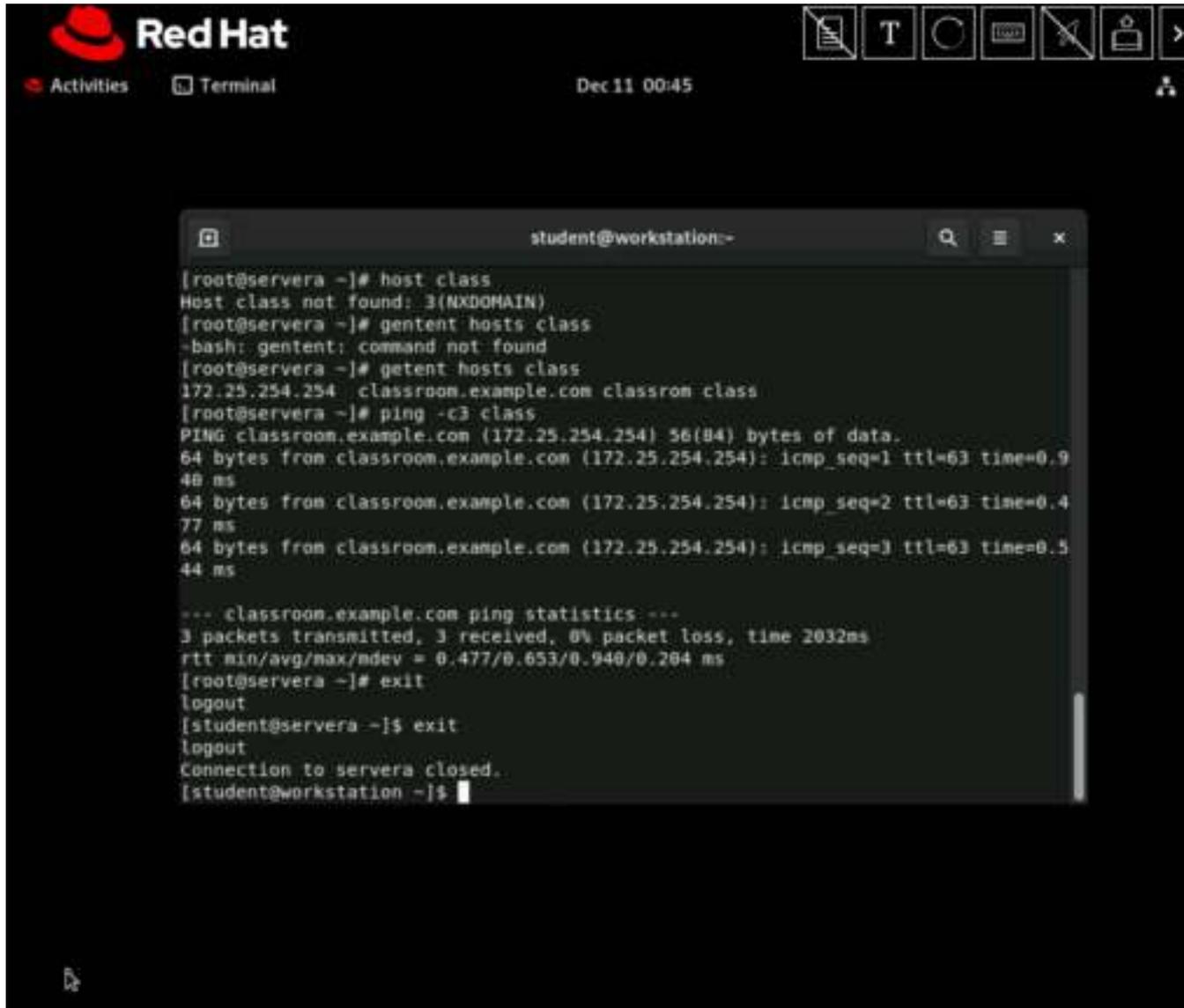
```
[root@servera ~]# host class  
Host class not found: 3(NXDOMAIN)  
[root@servera ~]# getent hosts class  
172.25.254.254 classroom.example.com classroom class
```

4. Use the `ping` command to send packets to the `class` server.

```
[root@servera ~]# ping -c3 class  
PING classroom.example.com (172.25.254.254) 56(84) bytes of data.  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=1 ttl=63  
time=1.21 ms  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=2 ttl=63  
time=0.688 ms  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=3 ttl=63  
time=0.559 ms  
  
--- classroom.example.com ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2046ms  
rtt min/avg/max/mdev = 0.559/0.820/1.214/0.283 ms
```

5. Return to the workstation system as the student user.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```



```
[root@servera ~]# host class
Host class not found: 3(NXDOMAIN)
[root@servera ~]# gentent hosts class
-bash: gentent: command not found
[root@servera ~]# getent hosts class
172.25.254.254 classroom.example.com classon class
[root@servera ~]# ping -c3 class
PING classroom.example.com (172.25.254.254) 56(84) bytes of data:
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=1 ttl=63 time=0.948 ms
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=2 ttl=63 time=0.477 ms
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=3 ttl=63 time=0.544 ms

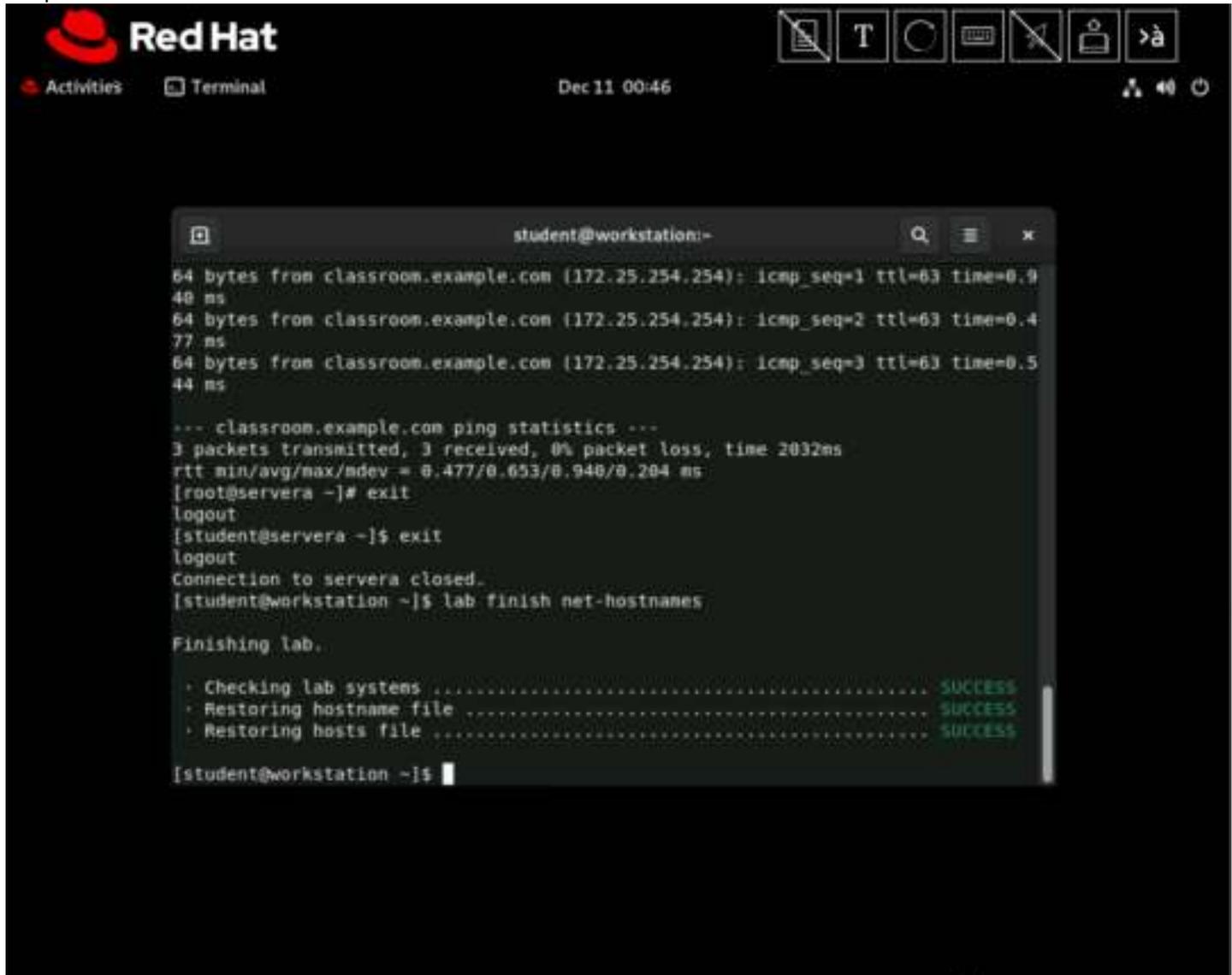
--- classroom.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.477/0.653/0.948/0.284 ms
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Finish

On the workstation machine, change to the `student` user home directory and use the `lab` command to complete this exercise. This step is important to ensure that resources from previous exercises do not impact upcoming exercises.

```
[student@workstation ~]$ lab finish net-hostnames
```

This concludes the section.



The screenshot shows a Red Hat desktop environment with a terminal window open. The terminal displays the output of a ping command to classroom.example.com (172.25.254.254) and the completion of a lab. The ping results show three successful packets with varying response times. The lab completion message indicates that the lab systems, hostname file, and hosts file were all restored successfully.

```
student@workstation:~  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=1 ttl=63 time=0.948 ms  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=2 ttl=63 time=0.477 ms  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=3 ttl=63 time=0.544 ms  
--- classroom.example.com ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2032ms  
rtt min/avg/max/mdev = 0.477/0.653/0.948/0.204 ms  
[root@servera ~]# exit  
logout  
[student@servera ~]# exit  
logout  
Connection to servera closed.  
[student@workstation ~]# lab finish net-hostnames  
Finishing lab.  
· Checking lab systems ..... SUCCESS  
· Restoring hostname file ..... SUCCESS  
· Restoring hosts file ..... SUCCESS  
[student@workstation ~]#
```

Lab: Manage Networking

In this lab, you configure networking settings on a Red Hat Enterprise Linux server.

Outcomes

- Configure two static IPv4 addresses for the primary network interface.

As the `student` user on the `workstation` machine, use the `lab` command to prepare your system for this exercise.

This command prepares your environment and ensures that all required resources are available.

```
[student@workstation ~]$ lab start net-review
```



Procedure 12.5. Instructions

1. Log in to the `serverb` machine as the `student` user. Switch to the `root` user.

The screenshot shows a Red Hat desktop environment with a terminal window open. The terminal displays the following commands and output:

```

root@serverb:~# lab start net-review
Starting lab.
  • Checking lab systems ..... SUCCESS
  • Backing up default nmcli profile ..... SUCCESS
  • Backing up hosts file ..... SUCCESS
  • Setting hosts.backup file as immutable ..... SUCCESS

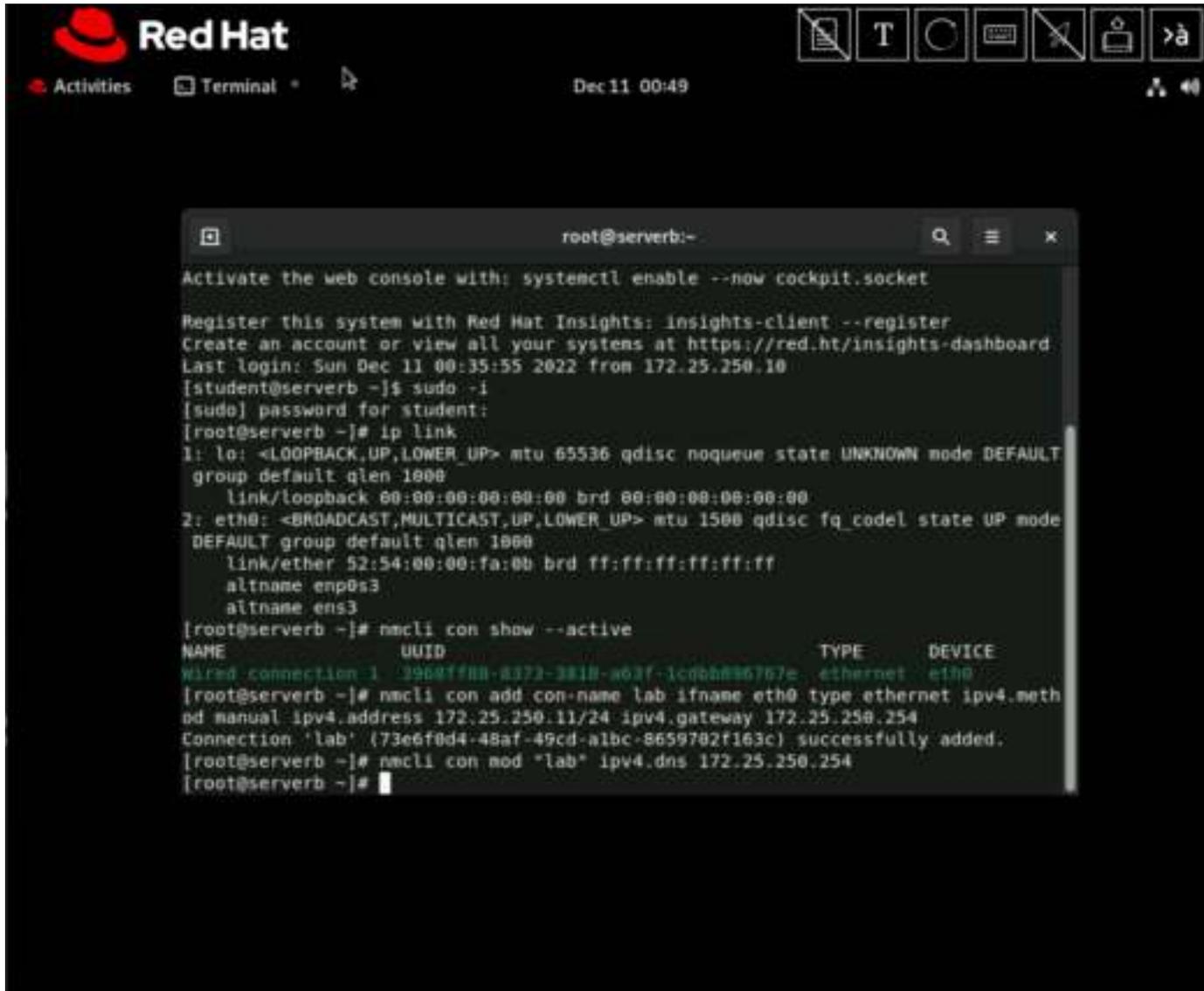
[student@workstation ~]$ ssh student@serverb
Activate the web console with: systemctl enable --now cockpit.socket

Register this system with Red Hat Insights: insights-client --register
Create an account or view all your systems at https://red.ht/insights-dashboard
Last login: Sun Dec 11 00:35:55 2022 from 172.25.250.10
[student@serverb ~]$ sudo -i
[sudo] password for student:
[root@serverb ~]#

```

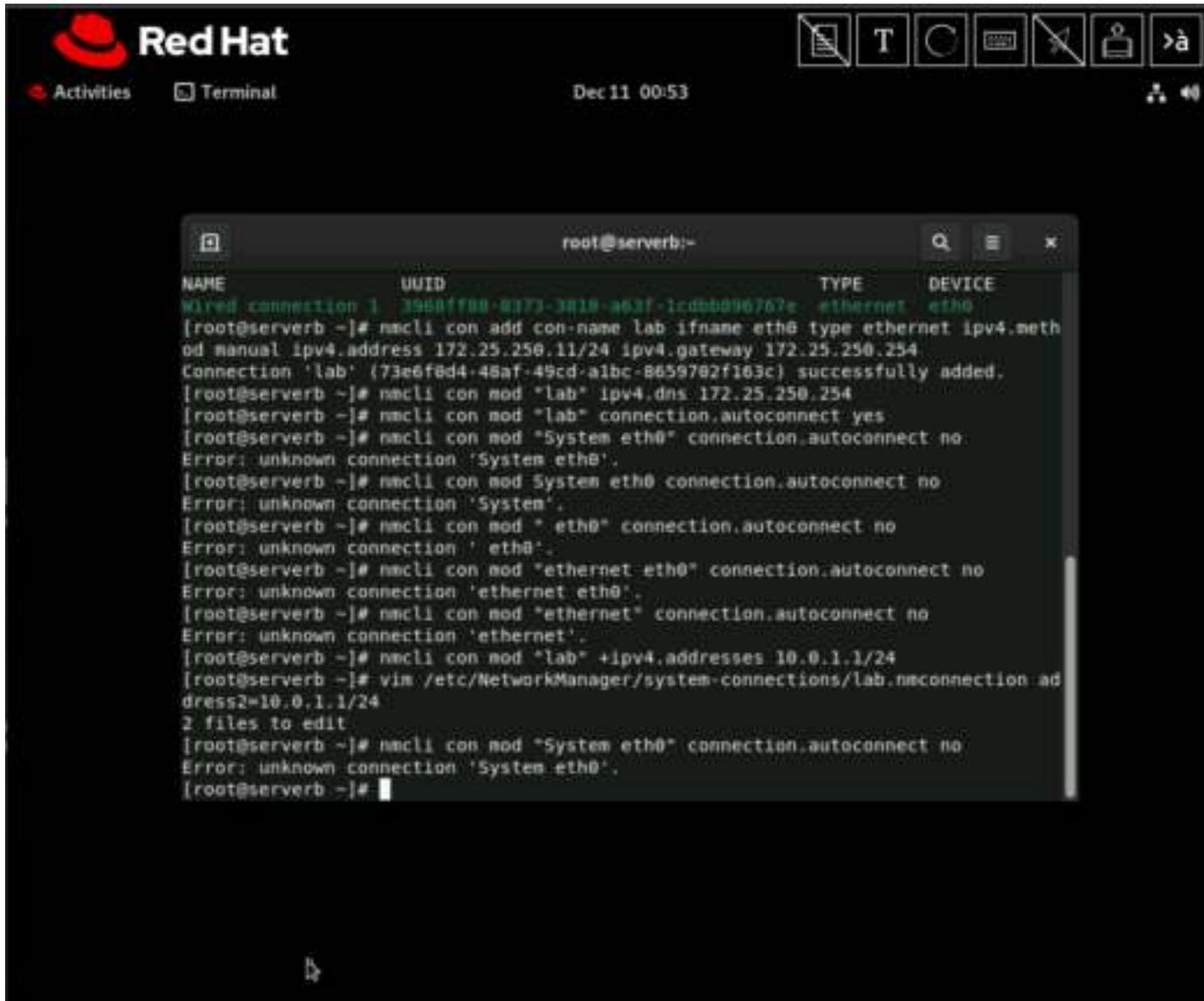
2. Create a connection with a static network configuration by using the settings in the table.

Parameter	Setting
Connection name	lab
Interface name	enX (might vary; use the interface with 52:54:00:00:fa:0b as its MAC address)
IP address	172.25.250.11/24
Gateway address	172.25.250.254
DNS address	172.25.250.254

A screenshot of a Red Hat Linux desktop environment. The top panel shows the Red Hat logo, 'Activities', 'Terminal', and the date 'Dec 11 00:49'. A terminal window is open, displaying the following commands and output:

```
root@serverb:~  
Activate the web console with: systemctl enable --now cockpit.socket  
Register this system with Red Hat Insights: insights-client --register  
Create an account or view all your systems at https://red.ht/insights-dashboard  
Last login: Sun Dec 11 00:35:55 2022 from 172.25.250.10  
[student@serverb ~]$ sudo -i  
[sudo] password for student:  
[root@serverb ~]# ip link  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT  
group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode  
DEFAULT group default qlen 1000  
    link/ether 52:54:00:00:fa:0b brd ff:ff:ff:ff:ff:ff  
    altname enp0s3  
    altname ens3  
[root@serverb ~]# nmcli con show --active  
NAME                UUID                                TYPE      DEVICE  
wired connection 1  29687f88-8372-3818-a63f-1c9bb886767e  ethernet eth0  
[root@serverb ~]# nmcli con add con-name lab ifname eth0 type ethernet ipv4.method manual ipv4.address 172.25.250.11/24 ipv4.gateway 172.25.250.254  
Connection 'lab' (73e6f0d4-48af-49cd-a1bc-8659702f163c) successfully added.  
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254  
[root@serverb ~]#
```

3. Configure the new connection to start automatically. Other connections should not start automatically.



The screenshot shows a terminal window on a Red Hat system. The terminal output displays the following commands and their results:

```
root@serverb:~# nmcli con add con-name lab ifname eth0 type ethernet ipv4.method manual ipv4.address 172.25.250.11/24 ipv4.gateway 172.25.250.254
Connection 'lab' (73e6fd4-48af-49cd-abc-8659702f163c) successfully added.
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254
[root@serverb ~]# nmcli con mod "lab" connection.autoconnect yes
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# nmcli con mod System eth0 connection.autoconnect no
Error: unknown connection 'System'.
[root@serverb ~]# nmcli con mod " eth0" connection.autoconnect no
Error: unknown connection ' eth0'.
[root@serverb ~]# nmcli con mod "ethernet eth0" connection.autoconnect no
Error: unknown connection 'ethernet eth0'.
[root@serverb ~]# nmcli con mod "ethernet" connection.autoconnect no
Error: unknown connection 'ethernet'.
[root@serverb ~]# nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
[root@serverb ~]# vim /etc/NetworkManager/system-connections/lab.nmconnection address2=10.0.1.1/24
2 files to edit
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]#
```

NAME	UUID	TYPE	DEVICE
Wired connection 1	3988ff80-8373-3818-a63f-1c0bb096767e	ethernet	eth0

4. Modify the new connection so that it also uses the IP address 10.0.1.1/24.



The screenshot shows a Red Hat Linux desktop environment. The top panel includes the Red Hat logo, the text "Red Hat", and system status icons. The main window is a terminal titled "root@serverb:~" displaying the configuration for a network connection named "lab". The configuration is as follows:

```
connection
id=lab
uuid=73e6f064-48af-49cd-a1bc-8659702f163c
type=ethernet
interface-name=eth0

[ethernet]

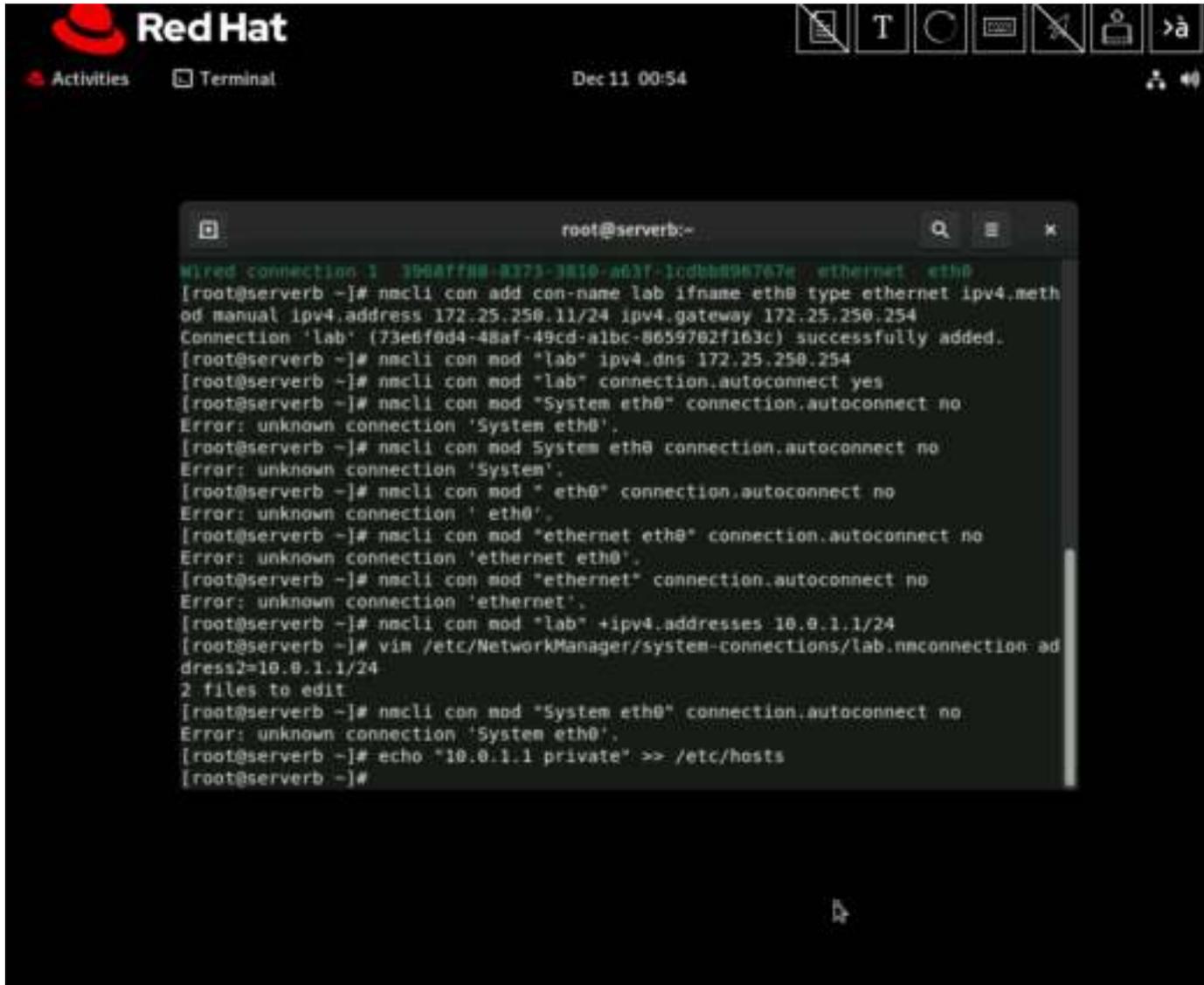
[ipv4]
address1=172.25.250.11/24,172.25.250.254
address2=10.0.1.1/24
dns=172.25.250.254;
method=manual

[ipv6]
addr-gen-mode=stable-privacy
method=auto

[proxy]
-
```

At the bottom of the terminal, a status bar shows the file path: `<ger/system-connections/lab.nmconnection*` with a size of 19L, 270B, and a cursor at 1,1.

5. Configure the `hosts` file so that you can reference the `10.0.1.1` IP address with the private name.



The screenshot shows a terminal window on a Red Hat system. The terminal output is as follows:

```
Wired connection 1 396aff88-8373-3810-a63f-1cddb898767e ethernet eth0
[root@serverb ~]# nmcli con add con-name lab ifname eth0 type ethernet ipv4.method manual ipv4.address 172.25.250.11/24 ipv4.gateway 172.25.250.254
Connection 'lab' (73e6f0d4-48af-49cd-a1bc-8659702f163c) successfully added.
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254
[root@serverb ~]# nmcli con mod "lab" connection.autoconnect yes
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# nmcli con mod System eth0 connection.autoconnect no
Error: unknown connection 'System'.
[root@serverb ~]# nmcli con mod "eth0" connection.autoconnect no
Error: unknown connection 'eth0'.
[root@serverb ~]# nmcli con mod "ethernet eth0" connection.autoconnect no
Error: unknown connection 'ethernet eth0'.
[root@serverb ~]# nmcli con mod "ethernet" connection.autoconnect no
Error: unknown connection 'ethernet'.
[root@serverb ~]# nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
[root@serverb ~]# vim /etc/NetworkManager/system-connections/lab.nmconnection-addresses=10.0.1.1/24
2 files to edit
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# echo "10.0.1.1 private" >> /etc/hosts
[root@serverb ~]#
```

6. Reboot the system.



```
Red Hat
Activities Terminal Dec 11 00:54

student@workstation:~
Connection 'lab' (73e6f0d4-48af-49cd-a1bc-8659702f163c) successfully added.
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254
[root@serverb ~]# nmcli con mod "lab" connection.autoconnect yes
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# nmcli con mod System eth0 connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# nmcli con mod "eth0" connection.autoconnect no
Error: unknown connection 'eth0'.
[root@serverb ~]# nmcli con mod "ethernet eth0" connection.autoconnect no
Error: unknown connection 'ethernet eth0'.
[root@serverb ~]# nmcli con mod "ethernet" connection.autoconnect no
Error: unknown connection 'ethernet'.
[root@serverb ~]# nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
[root@serverb ~]# vim /etc/NetworkManager/system-connections/lab.nmconnection ad
dress2=10.0.1.1/24
2 files to edit
[root@serverb ~]# nmcli con mod "System eth0" connection.autoconnect no
Error: unknown connection 'System eth0'.
[root@serverb ~]# echo "10.0.1.1 private" >> /etc/hosts
[root@serverb ~]# systemctl reboot
Connection to serverb closed by remote host.
Connection to serverb closed.
[student@workstation ~]#
```

7. Verify that the `serverb` machine is initialized.



The screenshot shows a Red Hat workstation desktop with a terminal window open. The terminal displays the following commands and output:

```
student@workstation:~$ nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
[student@workstation ~]$ nmcli con up lab
[student@workstation ~]$ nmcli con mod "System eth0" connection.autoconnect no
[student@workstation ~]$ echo "10.0.1.1 private" >> /etc/hosts
[student@workstation ~]$ systemctl reboot
[student@workstation ~]$ ping -c3 serverb
PING serverb.lab.example.com (172.25.250.11) 56(84) bytes of data:
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=1 ttl=64 time=0.754 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=2 ttl=64 time=0.486 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=3 ttl=64 time=0.331 ms

--- serverb.lab.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/ndev = 0.331/0.497/0.754/0.184 ms
[student@workstation ~]$
```

Evaluation

As the student user on the workstation machine, use the `lab` command to grade your work. Correct any reported failures and rerun the command until successful.

```
[student@workstation ~]$ lab grade net-review
```



The screenshot shows a Red Hat workstation interface with a terminal window open. The terminal displays the output of a ping command to serverb.lab.example.com (172.25.250.11) and the execution of the 'lab grade net-review' command. The ping statistics show 3 packets transmitted, 3 received, 0% packet loss, and a time of 2003ms. The lab grade report shows that all checks passed, resulting in an overall lab grade of PASS.

```
student@workstation:~$ ping -c 3 serverb.lab.example.com
486 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=3 ttl=64 time=0.331 ms

--- serverb.lab.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.331/0.497/0.754/0.184 ms
[student@workstation ~]$ lab grade net-review

Grading lab.

- Checking lab systems ..... SUCCESS
- Checking hosts file configuration ..... SUCCESS
- Checking that lab network connection exists ..... SUCCESS
- Checking that lab network connection is marked as active ..... SUCCESS
- Checking that lab autoconnect is marked yes ..... SUCCESS
- Checking lab IP address and netmask ..... SUCCESS
- Checking lab second IP address and netmask ..... SUCCESS
- Checking lab IPv4 gateway ..... SUCCESS
- Checking lab IPv4 DNS ..... SUCCESS

Overall lab grade: PASS
[student@workstation ~]$
```

Finish

On the `workstation` machine, change to the `student` user home directory and use the `lab` command to complete this exercise. This step is important to ensure that resources from previous exercises do not impact upcoming exercises.

```
[student@workstation ~]$ lab finish net-review
```



```
student@workstation:~$ lab finish net-review
- Checking that lab network connection exists ..... SUCCESS
- Checking that lab network connection is marked as active ..... SUCCESS
- Checking that lab autoconnect is marked yes ..... SUCCESS
- Checking lab IP address and netmask ..... SUCCESS
- Checking lab second IP address and netmask ..... SUCCESS
- Checking lab IPv4 gateway ..... SUCCESS
- Checking lab IPv4 DNS ..... SUCCESS

Overall lab grade: PASS

[student@workstation ~]$ lab finish net-review

Finishing lab.

- Checking lab systems ..... SUCCESS
- Removing autoconnection from current profile ..... SUCCESS
- Setting autoconnection for default profile ..... SUCCESS
- Setting up default profile ..... SUCCESS
- Removing lab connection profile ..... SUCCESS
- Unsetting hosts.backup file as immutable ..... SUCCESS
- Restoring hosts file ..... SUCCESS
- Removing backup default nmcli profile ..... SUCCESS

[student@workstation ~]$
```

This concludes the section.