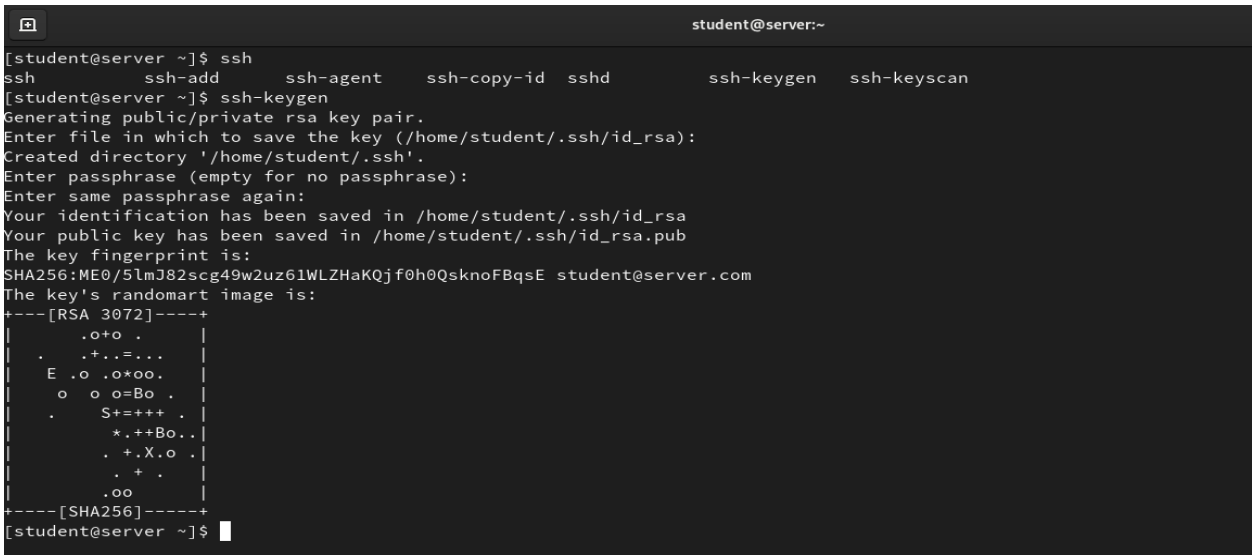
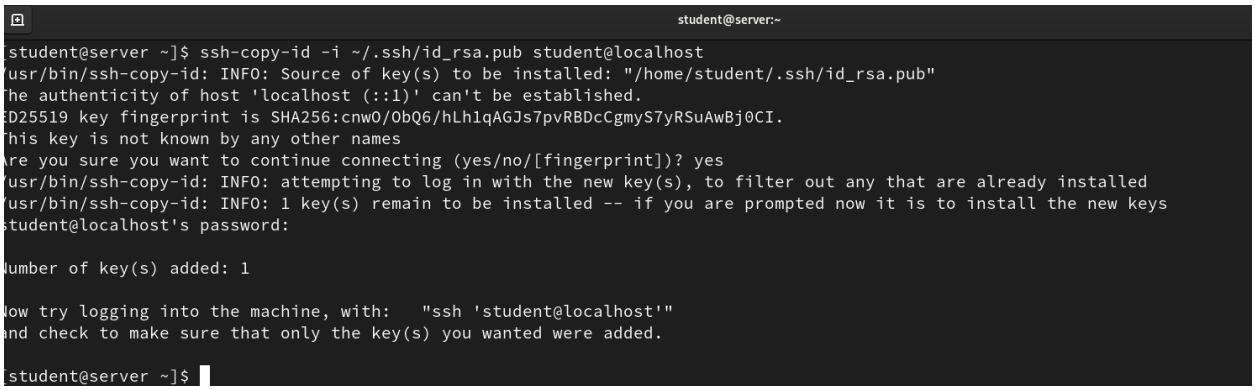
Task 1: Configuring and Securing SSH (Chapter 10 - Configuring and Securing SSH)

1- Configure SSH Key-Based Authentication

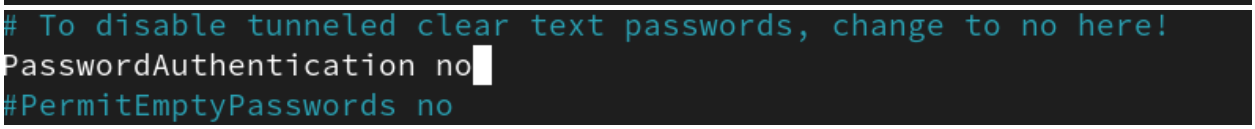
a- Generate an SSH key pair (RSA) for the user student.



b- Copy the public key to the server itself (localhost) to enable key-based authentication for this user.



c- Disable password-based SSH login for the student user by modifying the /etc/ssh/sshd_config file



d- Restart the sshd service and verify that you can log in using the key but not with a password.

```
[ibrahim@server ~]$ sudo systemctl restart sshd
[ibrahim@server ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-10-15 18:14:13 EEST; 13s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 3119 (sshd)
      Tasks: 1 (limit: 55007)
     Memory: 1.4M
        CPU: 19ms
     CGroup: /system.slice/sshd.service
             └─3119 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Oct 15 18:14:13 server.com systemd[1]: Starting OpenSSH server daemon...
Oct 15 18:14:13 server.com sshd[3119]: Server listening on 0.0.0.0 port 22.
Oct 15 18:14:13 server.com sshd[3119]: Server listening on :: port 22.
Oct 15 18:14:13 server.com systemd[1]: Started OpenSSH server daemon.
[ibrahim@server ~]$
[student@server ~]$ ssh student@localhost
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: Tue Oct 15 18:20:34 2024 from ::1
[student@server ~]$ exit
logout
Connection to localhost closed.
[student@server ~]$
```

2- Customizing SSH Configuration

a- Modify the SSH configuration to change the default SSH port from 22 to 2222.

```
[ibrahim@server ~]$ sudo nano /etc/ssh/sshd_config
[sudo] password for ibrahim:
```

```
Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

b- Restart the sshd service and verify that you can connect to the system using the new port.

```
ibrahim@server ~]$ sudo systemctl restart sshd
ibrahim@server ~]$ sudo systemctl status  sshd
 sshd.service - OpenSSH server daemon
    Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
    Active: active (running) since Tue 2024-10-15 16:47:27 EEST; 10s ago
      Docs: man:sshd(8)
            man:sshd_config(5)
  Main PID: 3978 (sshd)
     Tasks: 1 (limit: 55007)
    Memory: 1.4M
       CPU: 24ms
    CGroup: /system.slice/sshd.service
            └─3978 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

```
[student@server ~]$ ssh -p 2222 student@localhost
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 failed login: Tue Oct 15 18:00:38 EEST 2024 from login screen on seat0
There were 13 failed login attempts since the last successful login.
Last login: Tue Oct 15 16:48:14 2024
[student@server ~]$ exit
logout
Connection to localhost closed.
[student@server ~]$
```

c- Add a firewall rule to allow incoming SSH connections on port 2222. And ensure the change persists after reboot and verify the rule using firewall-cmd

```
[ibrahim@server ~]$ sudo firewall-cmd --permanent  --add-port=2222/tcp
[sudo] password for ibrahim:
Warning: ALREADY_ENABLED: 2222:tcp
success
[ibrahim@server ~]$ sudo firewall-cmd --reload
success
[ibrahim@server ~]$ sudo firewall-cmd --query-p
--query-panic      --query-port=      --query-protocol=
[ibrahim@server ~]$ sudo firewall-cmd --query-port=2222/tcp
yes
[ibrahim@server ~]$
```

>> this port was configured before in my server.

3- Securing SSH

a- Disable root login via SSH by editing the SSH configuration file.

```
ibrahim@server:~
[ibrahim@server ~]$ sudo nano /etc/ssh/sshd_config
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin no
```

b- Restart the sshd service and verify that the root user can no longer log in over SSH

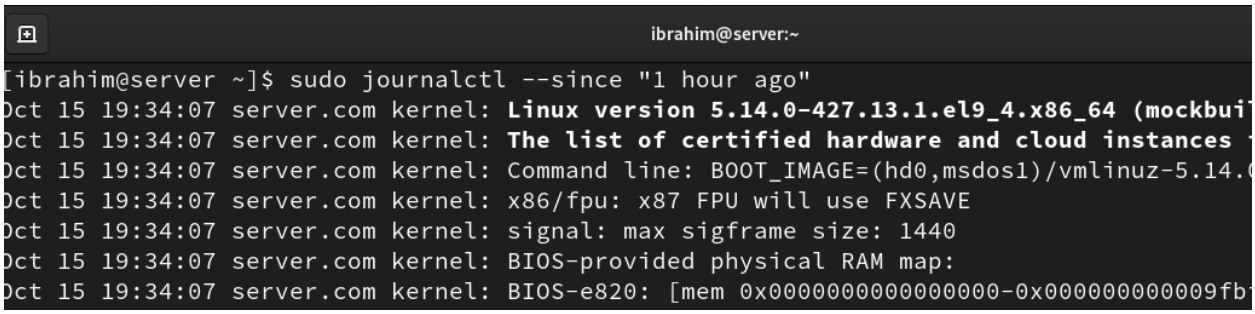
```
[ibrahim@server ~]$ sudo systemctl restart sshd
[sudo] password for ibrahim:
[ibrahim@server ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-10-15 17:06:00 EEST; 6s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 4479 (sshd)
      Tasks: 1 (limit: 55007)
     Memory: 1.4M
        CPU: 11ms
     CGroup: /system.slice/sshd.service
             └─4479 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Oct 15 17:06:00 server.com systemd[1]: Starting OpenSSH server daemon...
Oct 15 17:06:00 server.com sshd[4479]: Server listening on 0.0.0.0 port 2222.
Oct 15 17:06:00 server.com sshd[4479]: Server listening on :: port 2222.
Oct 15 17:06:00 server.com systemd[1]: Started OpenSSH server daemon.
[ibrahim@server ~]$
```

```
student@server:~
[student@server ~]$ ssh -p 2222 root@localhost
root@localhost: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[student@server ~]$
```

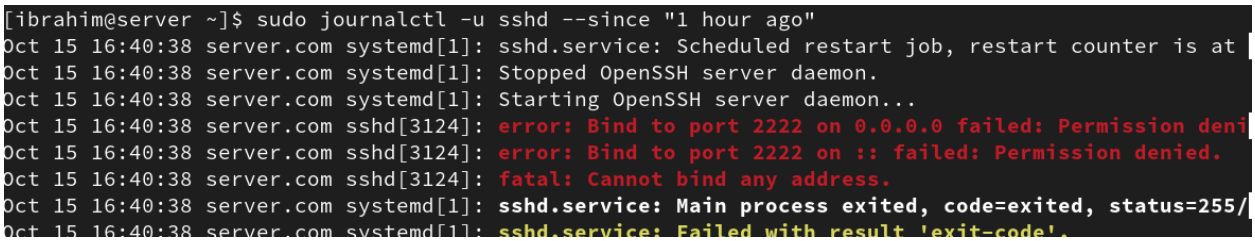
Task 2: Analyzing and Storing Logs (Chapter 11 - Analyzing and Storing Logs)

1- Review System Logs Using Journalctl

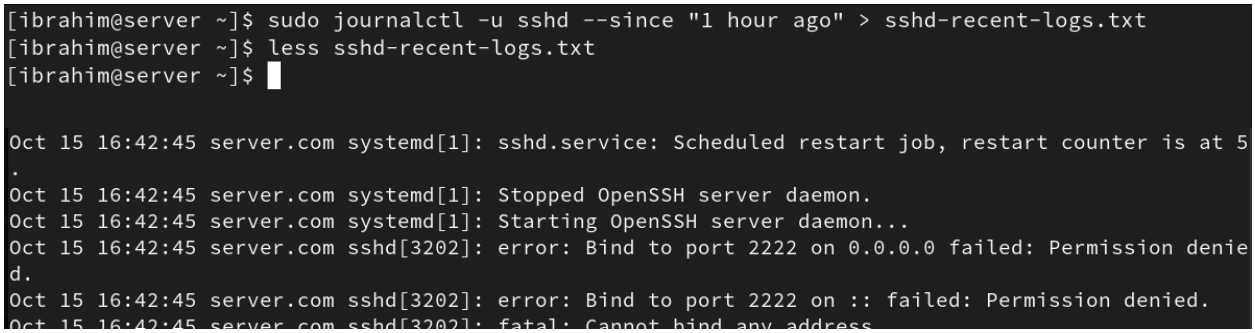
a- Use the journalctl command to display all logs generated in the last hour.



b- Filter the logs to show only messages related to the sshd service

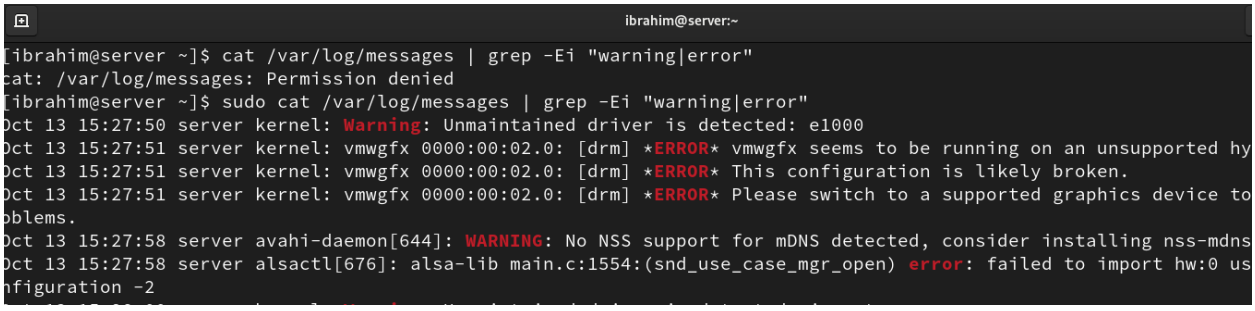


c- Save these filtered logs to a file named sshd-recent-logs.txt.

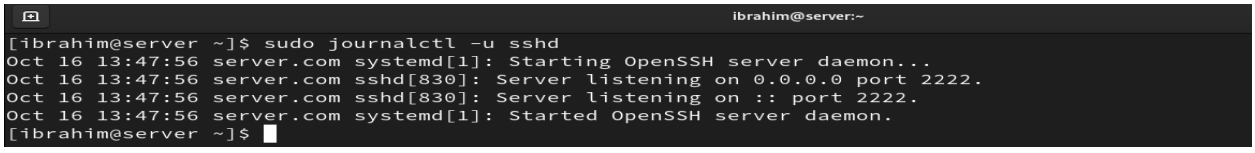


2- Working with Syslog

a- Open and review the contents of /var/log/messages to identify any warnings or errors.



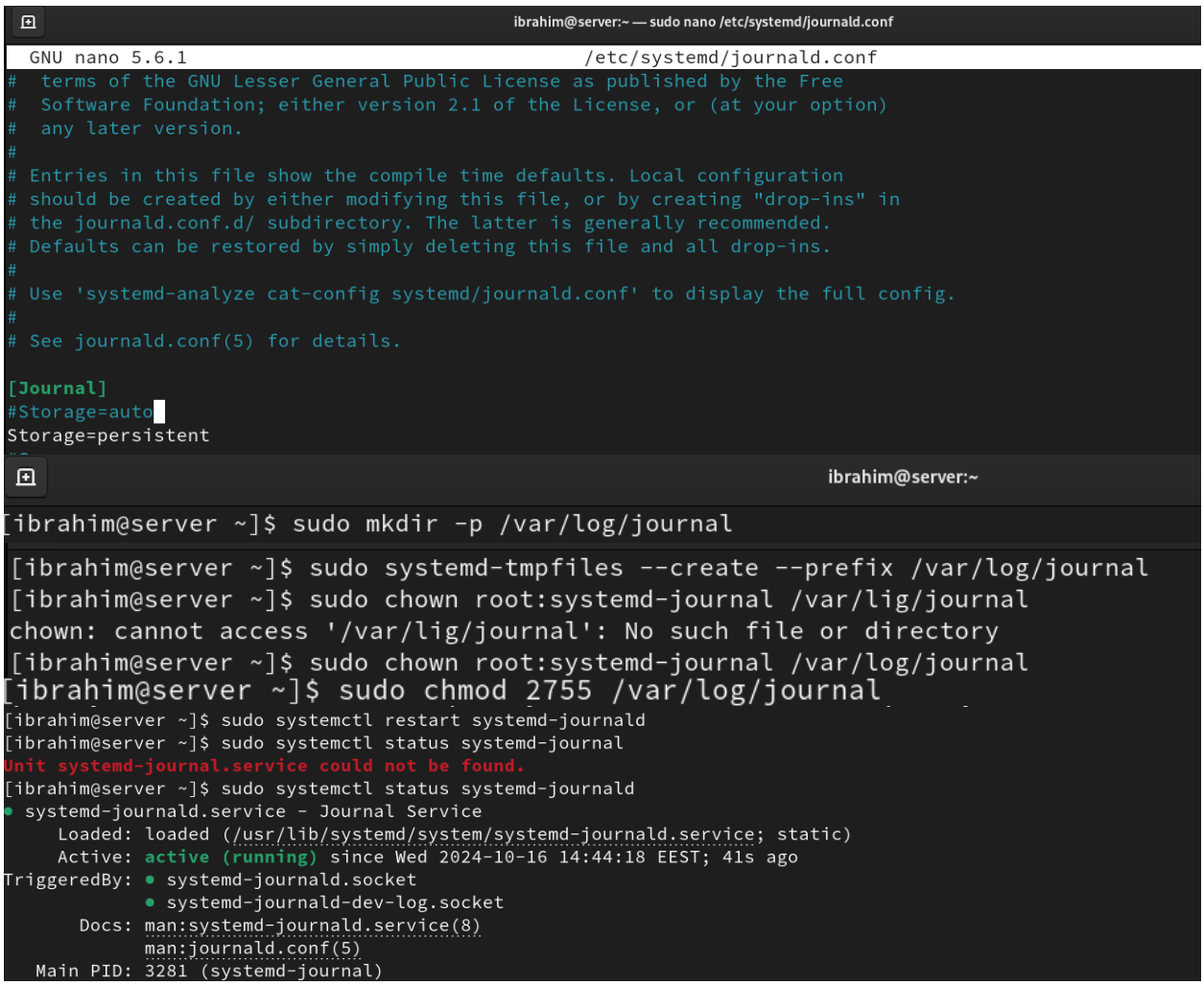
b- Identify any authentication failures (e.g., SSH login failures) in the /var/log/secure file and provide details of the failed login attempts



c- Save the failed login attempts to a file named failed-logins.txt.

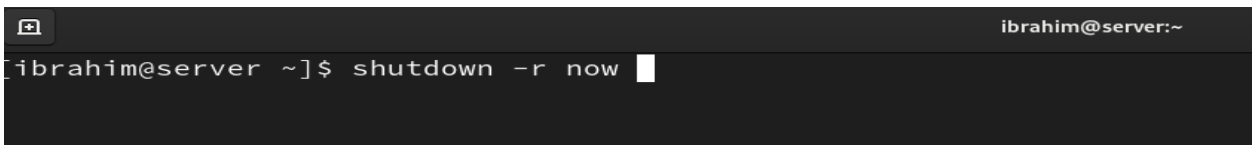
3- Preserving Logs with Systemd Journal:

a- Configure the systemd journal to persist logs across reboots by modifying the configuration in /etc/systemd/journald.conf.



>> then I will reboot my system.

b- Reboot the system and verify that logs are preserved.



```
ibrahim@server:~ — sudo journalctl
[ibrahim@server ~]$ sudo journalctl
Oct 16 17:54:07 server.com kernel: Linux version 5.14.0-427.13.1.el9_4.x86_64 (mockbuil
Oct 16 17:54:07 server.com kernel: The list of certified hardware and cloud instances f
Oct 16 17:54:07 server.com kernel: Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-5.14.0
Oct 16 17:54:07 server.com kernel: x86/fpu: x87 FPU will use FXSAVE
Oct 16 17:54:07 server.com kernel: signal: max sigframe size: 1440
Oct 16 17:54:07 server.com kernel: BIOS-provided physical RAM map:
```

c- Save the verification command output to a file named journal-persistence-check.txt.

```
ibrahim@server:~
[ibrahim@server ~]$ sudo grep "Storage" /etc/systemd/journald.conf > jounal-persist-check.txt
[ibrahim@server ~]$ cat jounal-persist-check.txt
#Storage=auto
Storage=persistent
[ibrahim@server ~]$
```

4- Monitoring Log Size

a- Use journalctl --disk-usage to check the size of the journal logs on the system

```
ibrahim@server:~
[ibrahim@server ~]$ sudo journalctl --disk-usage
Archived and active journals take up 16.0M in the file system.
[ibrahim@server ~]$
```

b- Set a limit of 200MB for the total journal size by configuring /etc/systemd/journald.conf.

```
#RateLimitBurst=10000
#SystemMaxUse=
SystemMaxUse=200M
```

```
[ibrahim@server ~]$ sudo nano /etc/systemd/journald.conf
[ibrahim@server ~]$ sudo systemctl restart journald
Failed to restart journald.service: Unit journald.service not found.
[ibrahim@server ~]$ sudo systemctl restart systemd-journald
[ibrahim@server ~]$ sudo systemctl status systemd-journald
● systemd-journald.service - Journal Service
     Loaded: loaded (/usr/lib/systemd/system/systemd-journald.service; static)
     Active: active (running) since Wed 2024-10-16 15:21:43 EEST; 26s ago
TriggeredBy: ● systemd-journald.socket
```

c- Verify that the setting is applied and working by using the appropriate journalctl command.

```
ibrahim@server:~
[ibrahim@server ~]$ sudo journalctl --disk-usage
Archived and active journals take up 16.0M in the file system.
[ibrahim@server ~]$ sudo cat /etc/systemd/journald.conf | grep "SystemMaxUse"
#SystemMaxUse=
SystemMaxUse=200M
[ibrahim@server ~]$
```

Task 3: Managing Networking (Chapter 12 - Managing Networking)

1- Checking Network Configuration

a- Display the current IP address, subnet mask, and gateway for your system using ip or nmcli.

```
ibrahim@server:~ — nmcli device show
[ibrahim@server ~]$ nmcli device show
GENERAL.DEVICE:                         enp0s3
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         08:00:27:FA:A6:84
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     enp0s3
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/2
WIRED-PROPERTIES.CARRIER:               on
IP4.ADDRESS[1]:                         192.168.1.11/24
IP4.GATEWAY:                            192.168.1.1
IP4.ROUTE[1]:                           dst = 192.168.1.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh = 192.168.1.1, mt = 100
IP4.DNS[1]:                             8.8.8.8
IP4.DNS[2]:                             8.8.4.4
IP4.DNS[3]:                             192.168.1.1
```

b- List all active network interfaces and their statuses

```
ibrahim@server:~
[ibrahim@server ~]$ nmcli device status
DEVICE  TYPE      STATE                   CONNECTION
enp0s3  ethernet  connected               enp0s3
lo      loopback  connected (externally)  lo
[ibrahim@server ~]$
```

c- Identify the default route and DNS servers configured for your system.

>> nmcli device show

```
IP4.GATEWAY:                            192.168.1.1
IP4.ROUTE[1]:                           dst = 192.168.1.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh = 192.168.1.1, mt = 100
IP4.DNS[1]:                             8.8.8.8
IP4.DNS[2]:                             8.8.4.4
IP4.DNS[3]:                             192.168.1.1
```

2- Configuring a Static IP Address

a- Create a new Interface to apply the following on it named ens33

>> can be added through the oracle VM

Settings > network > adapters > attach to your physical NIC

```
ibrahim@server:~
[ibrahim@server ~]$ nmcli device
DEVICE  TYPE      STATE                   CONNECTION
enp0s3  ethernet  connected               enp0s3
lo      loopback  connected (externally)  lo
enp0s8  ethernet  disconnected            --
[ibrahim@server ~]$
```

>> Now we need to connect this device

```
ibrahim@server:~
[ibrahim@server ~]$ nmcli device connect
enp0s3  enp0s8  help    lo
[ibrahim@server ~]$ nmcli device connect enp0s8
Device 'enp0s8' successfully activated with '17b21476-c668-4725-b271-d659fc7a841e'.
[ibrahim@server ~]$
```

>>renaming this interface to ens33

```
ibrahim@server:~
[ibrahim@server ~]$ sudo nmcli connection modify "enp0s8" connection.id ens33
[ibrahim@server ~]$ nmcli device
DEVICE  TYPE      STATE                   CONNECTION
enp0s3  ethernet  connected               enp0s3
enp0s8  ethernet  connected               ens33
lo      loopback  connected (externally)  lo
[ibrahim@server ~]$
```

b- Assign a static IP address 192.168.75.200/24 to the ens33 interface

```
ibrahim@server:~
[ibrahim@server ~]$ sudo nmcli connection modify ens33 ipv4.method manual ipv4.addresses 192.168.75.200/24
[ibrahim@server ~]$
```

c- Configure the system to use 192.168.75.1 as the default gateway and 8.8.8.8 as the DNS server.

```
ibrahim@server:~
[ibrahim@server ~]$ sudo nmcli connection modify ens33 ipv4.gateway 192.168.75.1 ipv4.dns 8.8.8.8
[ibrahim@server ~]$
```

d- Ensure the configuration persists across reboots by editing the appropriate network configuration files or using nmcli. e- Restart the networking service and verify the new IP address

>> in this solution I am using the nmcli which ensure that the configuration persist across reboots

>> now you can check the new configurations if it applied or not but first you need to shutdown the connection and up it again or it is recommended to restart the network manager

```
ibrahim@server:~
[ibrahim@server ~]$ sudo systemctl restart NetworkManager
[ibrahim@server ~]$ nmcli device show enp0s8
GENERAL.DEVICE:                         enp0s8
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         08:00:27:34:BB:37
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     ens33
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/3
WIRED-PROPERTIES.CARRIER:               on
IP4.ADDRESS[1]:                         192.168.75.200/24
IP4.GATEWAY:                            192.168.75.1
IP4.ROUTE[1]:                           dst = 0.0.0.0/0, nh = 192.168.75.1, mt = 102
IP4.ROUTE[2]:                           dst = 192.168.75.0/24, nh = 0.0.0.0, mt = 102
IP4.DNS[1]:                             8.8.8.8
IP6.ADDRESS[1]:                         fe80::99b9:879d:bf27:1b98/64
IP6.GATEWAY:                            --
IP6.ROUTE[1]:                           dst = fe80::/64, nh = ::, mt = 1024
IP6.DNS[1]:                             fe80::1
[ibrahim@server ~]$
```

3- Validating Network Configuration

a- Ping 192.168.1.1 (default gateway) and 8.8.8.8 to ensure network connectivity

```
ibrahim@server:~
[ibrahim@server ~]$ ping -c 5 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=73.8 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=2.46 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=2.42 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=2.83 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=3.64 ms

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4011ms
rtt min/avg/max/mdev = 2.424/17.037/73.833/28.401 ms
```

```
[ibrahim@server ~]$ ping  -c 5 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=61.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=61.1 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=60.7 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=60.8 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=115 time=61.3 ms

--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4020ms
rtt min/avg/max/mdev = 60.691/61.167/61.912/0.424 ms
```

b- Verify name resolution by pinging a domain name such as www.google.com to ensure DNS is functioning correctly.

```
ibrahim@server:~
[ibrahim@server ~]$ ping -c 5 www.google.com
PING www.google.com (142.250.200.228) 56(84) bytes of data.
64 bytes from mrs08s18-in-f4.1e100.net (142.250.200.228): icmp_seq=1 ttl=116 time=61.3 ms
64 bytes from mrs08s18-in-f4.1e100.net (142.250.200.228): icmp_seq=2 ttl=116 time=60.3 ms
64 bytes from mrs08s18-in-f4.1e100.net (142.250.200.228): icmp_seq=3 ttl=116 time=67.1 ms
64 bytes from mrs08s18-in-f4.1e100.net (142.250.200.228): icmp_seq=4 ttl=116 time=61.6 ms
64 bytes from mrs08s18-in-f4.1e100.net (142.250.200.228): icmp_seq=5 ttl=116 time=60.5 ms

--- www.google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4022ms
rtt min/avg/max/mdev = 60.321/62.149/67.052/2.500 ms
```

c- Use traceroute to track the path to www.redhat.com and save the output to a file named ne twork-trace.txt.

```
ibrahim@server:~
[ibrahim@server ~]$ tracepath www.redhat.com > network-trace.txt
[ibrahim@server ~]$ less network-trace.txt
```

```
ibrahim@server:~ — less network-trace.txt
 1?: [LOCALHOST]                      pmtu 1492
 1:  _gateway                                              4.247ms
 1:  _gateway                                              5.121ms
 2:  10.10.12.15                                          38.061ms
 3:  10.18.1.169                                          42.504ms asymm  4
 4:  10.64.13.133                                         39.145ms asymm  5
 5:  hu-0-0-0-5.br03.mrs01.as3491.net                     67.301ms asymm  6
 6:  no reply
 7:  be2780.ccr42.par01.atlas.cogentco.com                121.780ms asymm  9
```

4- Managing Hostnames

a- Set the hostname of your system to rhcsa_new.localdomain

```
ibrahim@rhcsanew:~
[ibrahim@rhcsanew ~]$ sudo hostnamectl hostname rhcsa_new.localdomain
[ibrahim@rhcsanew ~]$
```

b- Modify the /etc/hosts file to include the new hostname and its IP address

>>first you need to get your sever`s ip which is attached to the enp0s3 interface

192.168.1.11/24

```
[ibrahim@rhcsanew ~]$ ip addr show
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
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:fa:a6:84 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.11/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
```

>> Then you can add it in the /etc/hosts

```
ibrahim@rhcsanew:~
  GNU nano 5.6.1                                  /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.1.11          rhcsanew.localdomain
```

c- Verify the hostname has been correctly configured using the hostnamectl command.

```
ibrahim@rhcsanew:~
[ibrahim@rhcsanew ~]$ hostnamectl hostname
rhcsanew.localdomain
[ibrahim@rhcsanew ~]$
```

5- Network Troubleshooting

a- Simulate a network issue by bringing down the primary network interface (e.g., ens33) using nmcli or ip.

```
ibrahim@rhcsanew:~
[ibrahim@rhcsanew ~]$ sudo nmcli device down enp0s3
[sudo] password for ibrahim:
Device 'enp0s3' successfully disconnected.
[ibrahim@rhcsanew ~]$ nmcli device status
DEVICE  TYPE      STATE                   CONNECTION
enp0s8  ethernet  connected               ens33
lo      loopback  connected (externally)  lo
enp0s3  ethernet  disconnected            --
[ibrahim@rhcsanew ~]$
```

b- Attempt to ping an external IP like 8.8.8.8 and capture the error messages

```
[ibrahim@rhcsanew ~]$ ping -c 3 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 192.168.75.200 icmp_seq=1 Destination Host Unreachable
From 192.168.75.200 icmp_seq=2 Destination Host Unreachable
From 192.168.75.200 icmp_seq=3 Destination Host Unreachable

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2029ms
pipe 3
[ibrahim@rhcsanew ~]$
```

c- Bring the interface back up and verify network connectivity is restored.

```
ibrahim@rhcsanew:~
[ibrahim@rhcsanew ~]$ nmcli device up enp0s3
Device 'enp0s3' successfully activated with 'af4c267c-2dc9-3c79-8894-e5b2fea6dd2f'.
[ibrahim@rhcsanew ~]$ nmcli device
DEVICE  TYPE      STATE                   CONNECTION
enp0s3  ethernet  connected               enp0s3
enp0s8  ethernet  connected               ens33
lo      loopback  connected (externally)  lo
[ibrahim@rhcsanew ~]$ ping -c 3 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=65.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=60.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=61.9 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 60.732/62.835/65.854/2.189 ms
[ibrahim@rhcsanew ~]$
```

Task 4: Comprehensive Lab - SSH, Logs, and Networking

1- Comprehensive Scenario

a- As root, create a new user named netadmin and configure SSH key-based authentication for this user.

>> Create the user

```
root@rhcsanew:~
[root@rhcsanew ~]# useradd netadmin -s /bin/bash
[root@rhcsanew ~]# echo "netadmin:12345" | chpasswd
[root@rhcsanew ~]# tail -m 3 /etc/passwd
tail: invalid option -- 'm'
Try 'tail --help' for more information.
[root@rhcsanew ~]# tail -n 3 /etc/passwd
testuser:x:1006:1006::/home/testuser:/bin/bash
student:x:1007:1007::/home/student:/bin/bash
netadmin:x:1008:1008::/home/netadmin:/bin/bash
[root@rhcsanew ~]#
```

>> configure the ssh-key-based-auth(generate the keys)

```
[netadmin@rhcsanew ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/netadmin/.ssh/id_rsa):
Created directory '/home/netadmin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/netadmin/.ssh/id_rsa
Your public key has been saved in /home/netadmin/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Ipa7Wgtcvii/gBFsUbYpFN3HSCKPQztWPBEsCd2sUwQ netadmin@rhcsanew.localdomain
The key's randomart image is:
+---[RSA 3072]----+
|oBE%*.o          |
|+oX=Bo o         |
|.X.*. .          |
|o.*  .           |
|.  .= . S        |
|.o + o .         |
|o o +            |
|.. + +           |
| o=o+            |
+----[SHA256]-----+
```

>> add the public key in the authorized_keys file

```
netadmin@rhcsanew:~
[netadmin@rhcsanew ~]$ cat ~/.ssh/
id_rsa       id_rsa.pub   known_hosts
[netadmin@rhcsanew ~]$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
[netadmin@rhcsanew ~]$ ls ~/.ssh/
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[netadmin@rhcsanew ~]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC9fm40enH/HQTwiEUCyFHZoBmVjsQ97QGaU5TWgDucJSA4cZeugo8sM3iEryS2c0TcPur
mTKf3h9pKVxCr9MjJN05vtYnLEgG0GlvOvLm9hhK/iSjQw51Lj+01YRRJedbKvpo5A5d7xZfcaqek0+Bjdj4Nj9UJC4rq7HlX9G1Hx/w0Fu
qSVRveD5uzY8WfpSl9rhlP9reLYfxVdbFKMsg/TjQePVB6xM0UfPuK8FvbB6mwXuixSAYsvT9eZlY9VYnqN+epzd7u9/+9at14yu7uRhwEy
r3OkycMNuLhO3S9l6MtzbFywvIIT3PV2zj+EXc27U5N78zrBxOk1hbw0pbrf52mOKfEj0qXJts585lL0Vonizuv+FHgTzjxsDp/A85SDxzz
JIyhjokoBXn9baoVbo8sTlYRjBaxgUO8wKI7CNJJf+Pq07PnIq+0zakEZ7icRtvUYf2YA49L5WcotOjSm6xMI8k5DMl/CGNdur1Z4oUC8rX
yhh/NnYc3qc= netadmin@rhcsanew.localdomain
[netadmin@rhcsanew ~]$
```

>> change the permissions on the authorized_key file to 600

```
[root@rhcsanew ~]# usermod -aG wheel netadmin
[root@rhcsanew ~]# su - netadmin
[netadmin@rhcsanew ~]$ cd .ssh/
[netadmin@rhcsanew .ssh]$ sudo chmod 600 authorized_keys

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for netadmin:
[netadmin@rhcsanew .ssh]$ ls -l authorized_keys
-rw-------. 1 netadmin netadmin 583 Oct 16 18:57 authorized_keys
[netadmin@rhcsanew .ssh]$
```

b- Ensure that the netadmin user can only log in over SSH using port 2222.

>> I will need first to configure the sshd configuration file to allow the port 2222

>>restart the sshd service to ensure that everything works properly

>> then I will need to allow this port in the firewall and reload it to submit the configuration

```
netadmin@rhcsanew:~
  GNU nano 5.6.1                                  /etc/ssh/sshd_config
Include /etc/ssh/sshd_config.d/*.conf

# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
Port 2222
#AddressFamily any
```

>> close and save the changes then restart or reload the sshd service

```
[netadmin@rhcsanew ~]$ sudo nano /etc/ssh/sshd_config
[netadmin@rhcsanew ~]$ sudo systemctl restart sshd
[netadmin@rhcsanew ~]$ sudo systemctl status sshd
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-10-16 19:09:00 EEST; 9s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 10228 (sshd)
      Tasks: 1 (limit: 55007)
     Memory: 1.3M
        CPU: 12ms
     CGroup: /system.slice/sshd.service
             └─10228 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

>> allow this port in the firewall and reload it to submit the configuration

```
[netadmin@rhcsanew ~]$ sudo firewall-cmd --permanent --add-port=2222/tcp
Warning: ALREADY_ENABLED: 2222:tcp
success
[netadmin@rhcsanew ~]$ sudo firewall-cmd --re
--reload                              --remove-lockdown-whitelist-user=
--remove-forward                      --remove-masquerade
--remove-forward-port=                --remove-port=
--remove-icmp-block=                  --remove-protocol=
--remove-icmp-block-inversion         --remove-rich-rule=
--remove-interface=                   --remove-service=
--remove-lockdown-whitelist-command=  --remove-source=
--remove-lockdown-whitelist-context=  --remove-source-port=
--remove-lockdown-whitelist-uid=      --reset-to-defaults
[netadmin@rhcsanew ~]$ sudo firewall-cmd --reload
success
[netadmin@rhcsanew ~]$
```

c- Assign the netadmin user a static IP address 192.168.75.50/24 and configure DNS to use 1.1.1.1

```
[netadmin@rhcsanew ~]$ nmcli device
DEVICE  TYPE      STATE                   CONNECTION
enp0s3  ethernet  connected               enp0s3
enp0s8  ethernet  connected               ens33
lo      loopback  connected (externally)  lo
[netadmin@rhcsanew ~]$ nmcli connection modify ens33 ipv4.method manual ipv4.addresses 192.168.75.50/24 ipv4
.gateway 192.168.75.1 ipv4.dns 1.1.1.1
[netadmin@rhcsanew ~]$ nmcli device d
delete      disconnect  down
[netadmin@rhcsanew ~]$ nmcli device down enp0s8
Device 'enp0s8' successfully disconnected.
[netadmin@rhcsanew ~]$ nmcli device up enp0s8
Device 'enp0s8' successfully activated with '17b21476-c668-4725-b271-d659fc7a841e'.
[netadmin@rhcsanew ~]$
```

```
[netadmin@rhcsanew ~]$ nmcli device show enp0s8
GENERAL.DEVICE:                         enp0s8
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         08:00:27:34:BB:37
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     ens33
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/6
WIRED-PROPERTIES.CARRIER:               on
IP4.ADDRESS[1]:                         192.168.75.50/24
IP4.GATEWAY:                            192.168.75.1
IP4.ROUTE[1]:                           dst = 192.168.75.0/24, nh = 0.0.0.0, mt = 101
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh = 192.168.75.1, mt = 101
IP4.DNS[1]:                             1.1.1.1
```

d- Ensure the netadmin user's logs are saved to /var/log/netadmin-ssh-logs.txt every time they log in.

```
netadmin@rhcsanew:~
[netadmin@rhcsanew ~]$ sudo touch /var/log/netadmin-ss-log.txt
[netadmin@rhcsanew ~]$ ls -l /var/log/netadmin-ss-log.txt
-rw-r--r--. 1 root root 0 Oct 17 07:35 /var/log/netadmin-ss-log.txt
[netadmin@rhcsanew ~]$ sudo chown netadmin:netadmin /var/log/netadmin-ss-log.txt
[netadmin@rhcsanew ~]$ sudo chmod 644 /var/log/netadmin-ss-log.txt
[netadmin@rhcsanew ~]$ ls -l /var/log/netadmin-ss-log.txt
-rw-r--r--. 1 netadmin netadmin 0 Oct 17 07:35 /var/log/netadmin-ss-log.txt
[netadmin@rhcsanew ~]$
```

>> then add the following line showed below to .bashrc to store the log in actions every time the user netadmin logged on the system.

```
echo "$(date) - user netadmin logged in from $SSH_CLIENT" >> /var/log/netadmin-ss-log.txt
```

>> Here is the verification applied

```
[netadmin@rhcsanew ~]$ ssh -p 2222 netadmin@localhost
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: Thu Oct 17 07:31:22 2024
[netadmin@rhcsanew ~]$ sudo cat /var/log/netadmin-ss-log.txt
[sudo] password for netadmin:
Thu Oct 17 07:44:19 AM EEST 2024 - user netadmin logged in from ::1 35064 2222
[netadmin@rhcsanew ~]$
```