

# Chapter 12. Install and Update Software Packages

[Register Systems for Red Hat Support](#)

[Quiz: Register Systems for Red Hat Support](#)

[Explain and Investigate RPM Software Packages](#)

[Guided Exercise: Explain and Investigate RPM Software Packages](#)

[Install and Update Software Packages with DNF](#)

[Guided Exercise: Install and Update Software Packages with DNF](#)

[Enable DNF Software Repositories](#)

[Guided Exercise: Enable DNF Software Repositories](#)

[Lab: Install and Update Software Packages](#)

[Summary](#)

**Abstract**

<b>Goal</b>	Download, install, update, and manage software packages from Red Hat and DNF package repositories.
<b>Objectives</b>	<ul style="list-style-type: none"><li>• Register a system to your Red Hat account and assign it entitlements for software updates and support services with Red Hat Subscription Management.</li><li>• Explain how software is provided as RPM packages, and investigate the DNF and RPM installed system packages.</li><li>• Find, install, and update software packages with the <code>dnf</code> command.</li><li>• Enable and disable server use of Red Hat or third-party DNF repositories.</li></ul>
<b>Sections</b>	<ul style="list-style-type: none"><li>• Register Systems for Red Hat Support (and Quiz)</li><li>• Explain and Investigate RPM Software Packages (and Guided Exercise)</li><li>• Install and Update Software Packages with DNF (and Guided Exercise)</li><li>• Enable DNF Software Repositories (and Guided Exercise)</li></ul>
<b>Lab</b>	Install and Update Software Packages

## Register Systems for Red Hat Support

### Objectives

Register a system to your Red Hat account and assign it entitlements for software updates and support services with Red Hat Subscription Management.

## Red Hat Subscription Management

Red Hat Subscription Management provides tools to entitle machines to product subscriptions, for administrators to get updates to software packages and to track information about support contracts and subscriptions that the systems use. Standard tools such as the `dnf` command obtain software packages and updates through a content distribution network that the Red Hat Content Delivery Network provides.

You can perform the following main tasks with the Red Hat Subscription Management tools:

- *Register* a system to associate it with the Red Hat account with an active subscription. With the Subscription Manager, the system can register uniquely in the subscription service inventory. You can unregister the system when it is not in use.
- *Subscribe* a system to entitle it to updates for the selected Red Hat products. Subscriptions have specific levels of support, expiration dates, and default repositories. The tools help to either auto-attach or select a specific entitlement.
- *Enable repositories* to provide software packages. By default, each subscription enables multiple repositories; other repositories such as updates or source code are enabled or disabled. A repository is a central location for storing and maintaining software packages.
- *Review and track* available or consumed entitlements. In the Red Hat Customer Portal, you might view the subscription information locally on a specific system or for a Red Hat account.

### Simple Content Access

Simple Content Access (SCA) is a Red Hat subscription management capability. When you enable SCA for your organization, the entitlement process is simplified. SCA eliminates the requirement to attach subscriptions at a per-system level. You register your systems, enable the repositories that each system needs, and begin installing software packages.

Simple Content Access is an optional feature of Red Hat Satellite Server and Red Hat Subscription Management. This course includes the subscription commands, as needed, if you have not yet enabled SCA.

## Subscribe a System with Red Hat Subscription Manager

Different options exist to register a system with the Red Hat Customer Portal. For example, you can access a graphical interface by using a GNOME application or through the RHEL web console, or you can register your system by using a command-line tool.

To register a system by using a GNOME application, launch the Red Hat Subscription Manager application from the **Activities** menu. Type *subscription* in the **Type to search** field and click the **Red Hat Subscription Manager** application. When prompted, enter the appropriate password to authenticate. In the **Subscriptions** window, click **Register** to open the **Register System** dialog box.

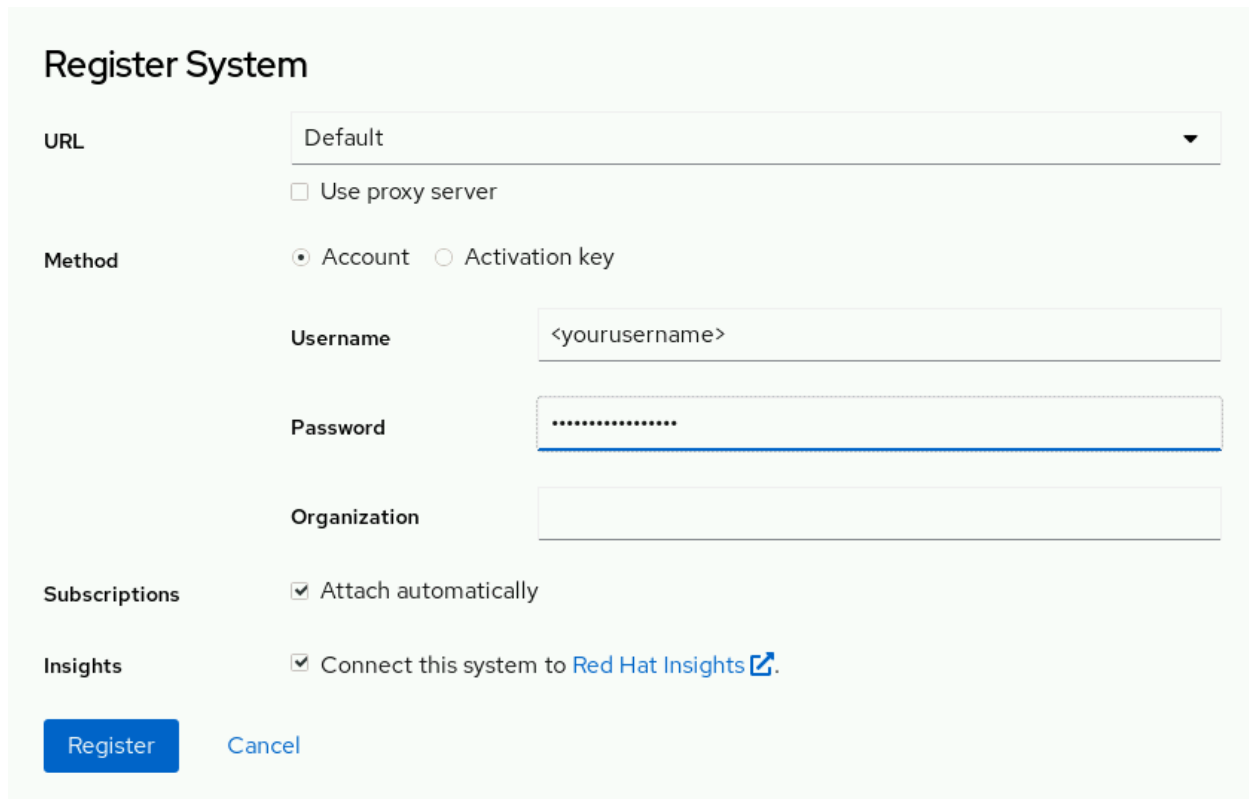
The image shows a 'Register System' dialog box with a light green background. At the top, the title 'Register System' is in bold. Below it, there are several sections: 'URL' with a dropdown menu set to 'Default' and a checkbox for 'Use proxy server'; 'Method' with radio buttons for 'Account' (selected) and 'Activation key'; 'Username' with a text field containing '<yourusername>'; 'Password' with a text field filled with dots; 'Organization' with an empty text field; 'Subscriptions' with a checked checkbox for 'Attach automatically'; and 'Insights' with a checked checkbox for 'Connect this system to Red Hat Insights' followed by a link icon. At the bottom left, there is a blue 'Register' button, and at the bottom right, there is a 'Cancel' link.

Figure 12.1: The Register System dialog box

By default, systems register to the Red Hat Customer Portal. Provide the login and the password for your Red Hat Customer Portal account and click **Register** to register the system. When registered, the system automatically attaches an available subscription.

Close the **Subscriptions** window after registering and assigning the system to a subscription. The system is now subscribed and ready to receive updates or to

install new software according to the subscription that is attached to the Red Hat Content Delivery Network.

## Subscribe a System with the RHEL Web Console

To register a system with the web console, you must log in as a privileged user. Click **Subscriptions** and then click **Register**:

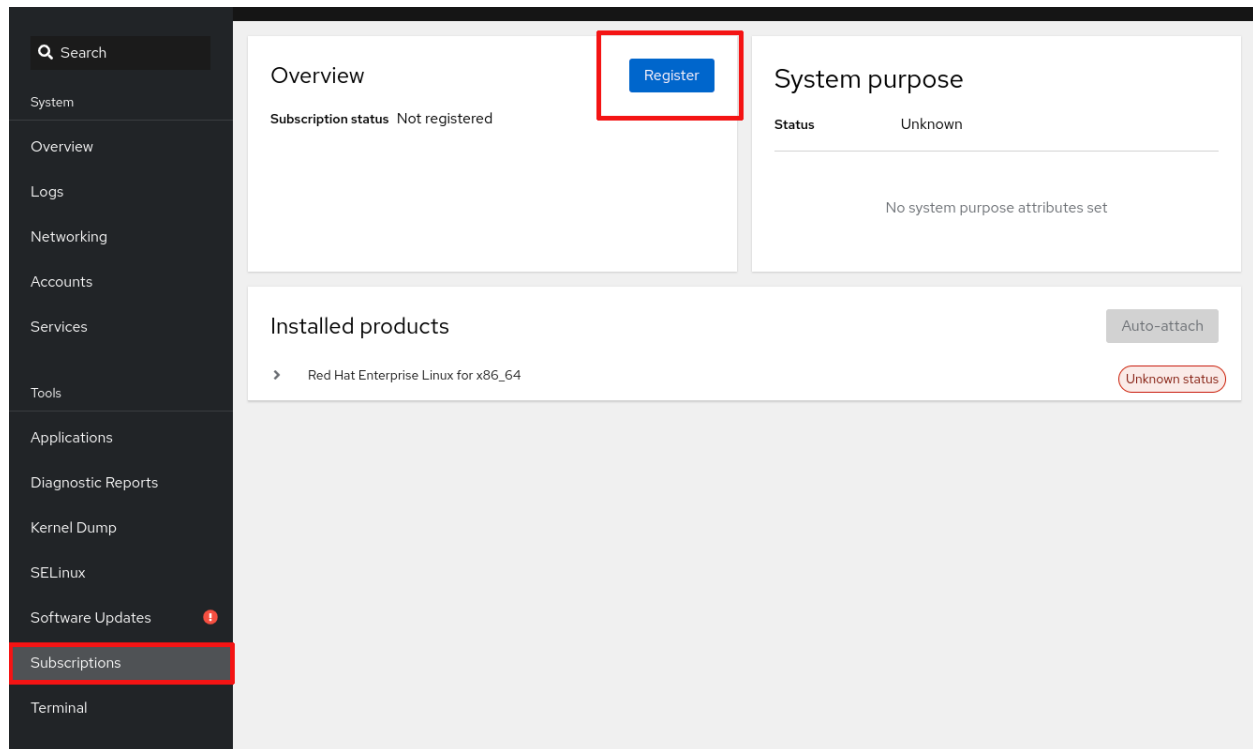


Figure 12.2: Web console subscriptions

The **Register System** dialog box in the web console is similar to the GNOME application. Provide the login and the password for your Red Hat Customer Portal account and click **Register** to register the system. You can then install new software or update your system, according to its attached subscription.

## Subscribe a System with the Command Line

Use the `subscription-manager` command to register a system without using a graphical environment. The `subscription-manager` command automatically attaches a system to the best-matched compatible subscriptions for the system.

Register a system by using the credentials of the Red Hat Customer Portal as the root user:

```
[root@host ~]# subscription-manager register --username <yourusername>
Registering to: subscription.rhsm.redhat.com:443/subscription
Password: yourpassword
The system has been registered with ID: 1457f7e9-f37e-4e93-960a-c94fe08e1b4f
The registered system name is: host.example.com
```

View available subscriptions for your Red Hat account:

```
[root@host ~]# subscription-manager list --available
-----
    Available Subscriptions
-----
...output omitted...
```

Auto-attach a subscription:

```
[root@host ~]# subscription-manager attach --auto
...output omitted...
```

Alternatively, attach a subscription from a specific pool from the list of available subscriptions:

```
[root@host ~]# subscription-manager attach --pool=poolID
...output omitted...
```

View consumed subscriptions:

```
[root@host ~]# subscription-manager list --consumed
...output omitted...
```

Unregister a system:

```
[root@host ~]# subscription-manager unregister
Unregistering from: subscription.rhsm.redhat.com:443/subscription
System has been unregistered.
```

Activation Keys

An *activation key* is a preconfigured subscription management file that is available for use with both Red Hat Satellite Server and subscription management through the Red Hat Customer Portal. Use the `subscription-manager` command with activation keys to simplify the registration and assignment of predefined subscriptions. This method of registration is beneficial for automating installations and deployments. For organizations that enable Simple Content Access, activation keys can register systems and enable repositories without needing to attach subscriptions.

## Entitlement Certificates

Digital certificates store current entitlement information on the local system. The registered system stores the entitlement certificates under the `/etc/pki` directory.

- `/etc/pki/product` certificates indicate installed Red Hat products.
- `/etc/pki/consumer` certificates identify the Red Hat account for registration.
- `/etc/pki/entitlement` certificates indicate which subscriptions are attached.

The `rct` command inspects the certificates, and the `subscription-manager` command examines the attached subscriptions on the system.

## References

`subscription-manager(8)` and `rct(8)` man pages

For further information, refer to *Registering the System and Managing Subscriptions* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/9/html-single/configuring\\_basic\\_system\\_settings/assembly\\_registering-the-system-and-managing-subscriptions\\_configuring-basic-system-settings](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/configuring_basic_system_settings/assembly_registering-the-system-and-managing-subscriptions_configuring-basic-system-settings)

[Next](#)

## Quiz: Register Systems for Red Hat Support

Choose the correct answer to the following questions:

1.

2.

1. Which item helps to register the system to Red Hat Subscription Management without a username and password?

A Organization ID

B Proxy URL

C Activation keys

D dnf

3. CheckResetShow Solution

4.

5.

2. Which GUI tool is used to register and subscribe a system?

A PackageKit

B gpk-application

C

Red Hat Subscription Manager

D

gnome-software

6. CheckResetShow Solution

7.

8.

3. Which directory stores the certificates for Red Hat products when using entitlement certificates?

A

/etc/pki/entitlement

B

/etc/subscription/product

C

/etc/pki/product

D

/etc/certs/pki

E

None of the previous options.

9. CheckResetShow Solution

[Previous](#) [Next](#)

## Explain and Investigate RPM Software Packages

### Objectives

Explain how software is provided as RPM packages, and investigate the DNF and RPM installed system packages.



## Software Packages and RPM

The RPM Package Manager, which Red Hat originally developed, provides a standard way to package software for distribution. Managing software in the form of RPM packages is simpler than working with software that is extracted to a file system from an archive. With RPM packages, administrators can track which files the software package installs, which files the software package removes if you uninstall it, and it verifies that supporting packages are present when you install it. The local RPM database on your system stores the information about installed packages. Red Hat provides all software for Red Hat Enterprise Linux as an RPM package.

RPM package file names consist of four elements (plus the `.rpm` suffix): `name-version-release.architecture:`



Figure 12.3: RPM file name elements

- NAME is one or more words to describe the contents (coreutils).
- VERSION is the version number of the original software (8.32).
- RELEASE is the release number of the package based on that version, and is set by the packager, who might not be the original software developer (31.e19).
- ARCH is the processor architecture that the package is compiled to run on. The x86\_64 value indicates that this package is built for the 64-bit version of the x86 instruction set (as opposed to aarch64 for 64-bit ARM, and so on).

RPM packages are often downloaded from repositories. A repository is a central location for storing and maintaining RPM software packages.

You require only the package name to install RPM packages from repositories.

- If multiple versions exist, then the RPM Package Manager installs the package with the later version number.
- If multiple releases of a single version exist, then the RPM Package Manager installs the package with the later release number.

Each RPM package is an archive with the following components:

- The files that the package installs in your system.
- Information about the package (metadata), such as the name, version, release, and architecture; a summary and description of the package; whether it requires other packages to be installed; licensing; a package change log; and other details.
- Scripts that might run when you install, update, or remove the package. These scripts might also run when you install, update, or remove other packages.

Typically, software providers digitally sign RPM packages with GPG (GNU Privacy Guard) keys. (Red Hat digitally signs all packages that it releases.) The RPM system verifies package integrity by confirming that the package is signed with the appropriate GPG key. The RPM system fails to install a package if the GPG signature does not match.

## Update Software with RPM Packages

Red Hat generates a complete RPM package to update software. An administrator who installs that package gets only the most recent version of the package. You do not need to install an earlier version of a package to patch it. To update software,

RPM removes the earlier version of the package and installs the latest version. Updates usually retain configuration files, but the packager of the new version defines the exact behavior.

Typically, only one version of a package is installed at a time. If a package is built with non-conflicting file names, then you might install multiple versions.

The `kernel` package is an example of installing multiple package versions. Because you test a new kernel only by booting to that kernel, the package is designed to support installing multiple versions. If the new kernel fails to boot, then you can revert to the previous kernel.

## Examine RPM Packages

The `rpm` utility is a low-level tool that can retrieve information about the contents of package files and installed packages. By default, the tool gets information from a local database of installed packages. Use the `rpm` command `-p` option to get information about a downloaded but uninstalled package file. Use this option to inspect the package contents before installing.

Retrieve general information about installed packages:

- **`rpm -qa`** : List all installed packages.
- **`rpm -qf FILENAME`** : Determine which package provides *FILENAME*.

```
[user@host ~]$ rpm -qf /etc/yum.repos.d
redhat-release-9.1-1.0.el9.x86_64
```

Get information about specific packages:

- **`rpm -q`** : List the currently installed package version.

```
[user@host ~]$ rpm -q dnf
dnf-4.10.0-4.el9.noarch
```

- **`rpm -qi`** : Get detailed package information.
- **`rpm -ql`** : List the files that the package installs.

```
[user@host ~]$ rpm -ql dnf
/usr/bin/dnf
/usr/lib/systemd/system/dnf-makecache.service
```

```
/usr/lib/systemd/system/dnf-makecache.timer
/usr/share/bash-completion
/usr/share/bash-completion/completions
/usr/share/bash-completion/completions/dnf
...output omitted...
```

- **rpm -qc** : List only the configuration files that the package installs.

```
[user@host ~]$ rpm -qc openssh-clients
/etc/ssh/ssh_config
/etc/ssh/ssh_config.d/50-redhat.conf
```

- **rpm -qd** : List only the documentation files that the package installs.

```
[user@host ~]$ rpm -qd openssh-clients
/usr/share/man/man1/scp.1.gz
/usr/share/man/man1/sftp.1.gz
/usr/share/man/man1/ssh-add.1.gz
/usr/share/man/man1/ssh-agent.1.gz
...output omitted...
```

- **rpm -q --scripts** : List the shell scripts that run before or after you install or remove the package.

```
[user@host ~]$ rpm -q --scripts openssh-server
preinstall scriptlet (using /bin/sh):
getent group sshd >/dev/null || groupadd -g 74 -r sshd || :
getent passwd sshd >/dev/null || \
    useradd -c "Privilege-separated SSH" -u 74 -g sshd \
    -s /sbin/nologin -r -d /usr/share/empty.sshd sshd 2> /dev/null || :
postinstall scriptlet (using /bin/sh):

if [ $1 -eq 1 ] && [ -x "/usr/lib/systemd/systemd-update-helper" ]; then
    # Initial installation
    /usr/lib/systemd/systemd-update-helper install-system-units sshd.service sshd.soc
ket || :
```

```
fi
```

```
...output omitted...
```

- **rpm -q --changelog** : List the change log information for the package.

```
[user@host ~]$ rpm -q --changelog audit
```

```
* Tue Feb 22 2022 Sergio Correia <scorreia@redhat.com> - 3.0.7-101
```

```
- Adjust sample-rules dir permissions
```

```
Resolves: rhbz#2054432 - /usr/share/audit/sample-rules is no longer readable by non-root users
```

```
* Tue Jan 25 2022 Sergio Correia <scorreia@redhat.com> - 3.0.7-100
```

```
- New upstream release, 3.0.7
```

```
Resolves: rhbz#2019929 - capability=unknown-capability(39) in audit messages
```

```
...output omitted...
```

Query local package files:

- **rpm -qlp** : List the files that the local package installs.

```
[user@host ~]$ ls -l podman-4.0.0-6.el9.x86_64.rpm
```

```
-rw-r--r--. 1 student student 13755101 Mar 22 11:35 podman-4.0.0-6.el9.x86_64.rpm2637-15.el9.x86_64.rpm
```

```
[user@host ~]$ rpm -qlp podman-4.0.0-6.el9.x86_64.rpm
```

```
/etc/cni/net.d
```

```
/etc/cni/net.d/87-podman-bridge.conflist
```

```
/usr/bin/podman
```

```
...output omitted...
```

## Install RPM Packages

Use the `rpm` command to install an RPM package that you downloaded to your local directory.

```
[root@host ~]# rpm -ivh podman-4.0.0-6.el9.x86_64.rpm
```

```
Verifying... ##### [100%]
```

```
Preparing... ##### [100%]
```

Updating / installing...

podman-2:4.0.0-6

##### [100%]

## Warning

Be careful when installing packages from third parties, not just because of the software that the packages might install, but because the RPM package might include arbitrary scripts that run as the `root` user as part of the installation process.

## Extracting RPM packages

Use the `rpm2cpio` command to extract files from an RPM package file without installing the package.

The `rpm2cpio` command converts an RPM package to a `cpio` archive. After the RPM package is converted to a `cpio` archive, the `cpio` command can extract a list of files.

Use the `cpio` command with the `-i` option to extract files from standard input. Use the `-d` option to create subdirectories as needed, starting in the current working directory. Use the `-v` option for verbose output.

```
[user@host tmp-extract]$ rpm2cpio httpd-2.4.51-7.el9_0.x86_64.rpm | cpio -idv
./etc/httpd/conf
./etc/httpd/conf.d/autoindex.conf
./etc/httpd/conf.d/userdir.conf
./etc/httpd/conf.d/welcome.conf
./etc/httpd/conf.modules.d
./etc/httpd/conf.modules.d/00-base.conf
./etc/httpd/conf.modules.d/00-dav.conf
./etc/httpd/conf.modules.d/00-mpm.conf
./etc/httpd/conf.modules.d/00-optional.conf
./etc/httpd/conf.modules.d/00-proxy.conf
./etc/httpd/conf.modules.d/00-systemd.conf
./etc/httpd/conf.modules.d/01-cgi.conf
./etc/httpd/conf.modules.d/README
./etc/httpd/conf/httpd.conf
...output omitted...
```

9774 blocks

```
[user@host tmp-extract]$ ls -l
```

total 1552

```
drwxr-xr-x. 5 user user      55 Feb  3 15:06 etc
-rw-r--r--. 1 user user 1588633 Feb  3 15:06 httpd-2.4.51-7.el9_0.x86_64.rpm
drwxr-xr-x. 3 user user      19 Feb  3 15:06 run
drwxr-xr-x. 7 user user      70 Feb  3 15:06 usr
drwxr-xr-x. 5 user user      41 Feb  3 15:06 var
```

Extract individual files by specifying the path of the file:

```
[user@host ~]$ rpm2cpio httpd-2.4.51-7.el9_0.x86_64.rpm | cpio -id "*/etc/httpd/conf/httpd.conf"
```

9774 blocks

```
[user@host ~]$ ls etc/httpd/conf/
httpd.conf
```

Use the `rpm2cpio` and `cpio -t` commands to list the files in an RPM package. Use the `-v` option of the `cpio` command for verbose output.

```
[student@servera ~]$ rpm2cpio httpd-2.4.51-7.el9_0.x86_64.rpm | cpio -tv
```

```
drwxr-xr-x  1 root  root           0 Mar 21 2022 ./etc/httpd/conf
-rw-r--r--  1 root  root      2893 Mar 21 2022 ./etc/httpd/conf.d/autoindex.conf
-rw-r--r--  1 root  root      1252 Mar 21 2022 ./etc/httpd/conf.d/userdir.conf
-rw-r--r--  1 root  root       653 Mar 21 2022 ./etc/httpd/conf.d/welcome.conf
drwxr-xr-x  1 root  root           0 Mar 21 2022 ./etc/httpd/conf.modules.d
-rw-r--r--  1 root  root      3372 Mar 21 2022 ./etc/httpd/conf.modules.d/00-base.conf
-rw-r--r--  1 root  root       139 Mar 21 2022 ./etc/httpd/conf.modules.d/00-dav.conf
-rw-r--r--  1 root  root       948 Mar 21 2022 ./etc/httpd/conf.modules.d/00-mpm.conf
-rw-r--r--  1 root  root       787 Mar 21 2022 ./etc/httpd/conf.modules.d/00-optional.conf
-rw-r--r--  1 root  root      1073 Mar 21 2022 ./etc/httpd/conf.modules.d/00-proxy.conf
```



```
-rw-r--r-- 1 root root      88 Mar 21 2022 ./etc/httpd/conf.modules.d/00-  
systemd.conf  
-rw-r--r-- 1 root root     367 Mar 21 2022 ./etc/httpd/conf.modules.d/01-  
cgi.conf  
-rw-r--r-- 1 root root     496 Mar 21 2022 ./etc/httpd/conf.modules.d/REA  
DME  
-rw-r--r-- 1 root root    12005 Mar 21 2022 ./etc/httpd/conf/httpd.conf  
...output omitted...  
9774 blocks
```

## References

rpm(8), rpm2cpio(8), cpio(1), and rpmkeys(8) man pages

[Previous](#) [Next](#)

## Guided Exercise: Explain and Investigate RPM Software Packages

In this exercise, you gather information about a package from a third party, extract files from it for inspection, and then install it on a server.

### Outcomes

- Install on a server a package that is not from the software repositories.

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 software-rpm
```

### Instructions

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

2. [student@workstation ~]\$ **ssh student@servera**
3. ...output omitted...

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

4. View package information and list files in the rhcsa-script-1.0.0-1.noarch.rpm package. Also view the script that runs when you install or uninstall the package.
  1. View information for the rhcsa-script-1.0.0-1.noarch.rpm package.

```
2. [student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -i
3. Name           : rhcsa-script
4. Version        : 1.0.0
5. Release        : 1
6. Architecture   : noarch
7. Install Date   : (not installed)
8. Group          : System
9. Size           : 593
10. License       : GPL
11. Signature     : (none)
12. Source RPM    : rhcsa-script-1.0.0-1.src.rpm
13. Build Date    : Wed 23 Mar 2022 08:24:21 AM EDT
14. Build Host    : localhost
15. Packager      : Bernardo Gargallo
16. URL           : http://example.com
17. Summary       : RHCSA Practice Script
18. Description   :
19. A RHCSA practice script.
```

```
The package changes the motd.
```

## Note

The preceding package modifies the *MOTD*, or "Message of the Day". A system displays the MOTD to users as they log in to systems.

20. List files in the rhcsa-script-1.0.0-1.noarch.rpm package.

```
21. [student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -l
```

```
/opt/rhcsa-script/mymotd
```

22. View the script that runs when you install or uninstall the rhcsa-script-1.0.0-1.noarch.rpm package.

```
23. [student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm --scripts
```

```
24. preinstall scriptlet (using /bin/sh):
```

```
25. if [ "$1" == "2" ]; then
```

```
26.     if [ -e /etc/motd.orig ]; then
```

```
27.         mv -f /etc/motd.orig /etc/motd
```

```
28.     fi
```

```
29. fi
```

```
30. postinstall scriptlet (using /bin/sh):
```

```
...output omitted...
```

5. Extract the contents of the rhcsa-script-1.0.0-1.noarch.rpm package to the /home/student directory.

1. Use the rpm2cpio and cpio -tv commands to list the files in the rhcsa-script-1.0.0-1.noarch.rpm package.

```
2. [student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -tv
```

```
3. -rw-r--r--  1 root    root          593 Mar 23 08:24 ./opt/rhcsa-script/
   mymotd
```

```
2 blocks
```

4. Extract all files from the rhcsa-script-1.0.0-1.noarch.rpm package to the /home/student directory. Use the rpm2cpio and cpio -idv commands to extract the files and create the parent directories where needed in verbose mode.

```
5. [student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -idv
```

```
6. ./opt/rhcsa-script/mymotd
```

```
2 blocks
```

7. List the files in the /home/student/opt directory to verify that the extracted files are the same as the files inside the package.

```
8. [student@servera ~]$ ls -lR opt
9. opt:
10. total 0
11. drwxr-xr-x. 2 student student 20 Mar 23 09:22 rhcsa-script
12.
13. opt/rhcsa-script:
14. total 4
```

```
-rw-r--r--. 1 student student 593 Mar 23 09:22 mymotd
```

6. Install the rhcsa-script-1.0.0-1.noarch.rpm package. Use the sudo command to gain superuser privileges to install the package.

1. Use the sudo rpm -ivh command to install the rhcsa-script-1.0.0-1.noarch.rpm RPM package.

```
2. [student@servera ~]$ sudo rpm -ivh rhcsa-script-1.0.0-1.noarch.rpm
3. [sudo] password for student: student
4. Verifying... ##### [1
  00%]
5. Preparing... ##### [1
  00%]
6. Updating / installing...
7. 1:rhcsa-script-1.0.0-1 ##### [1
  00%]
```

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

8. Use the rpm command to verify that you correctly installed the package.

```
9. [student@servera ~]$ rpm -q rhcsa-script
```

```
rhcsa-script-1.0.0-1.noarch
```

7. Exit from the servera machine and connect again to test the new message of the day.

```
8. [student@servera ~]$ exit
```

```
9. logout
```

10. Connection to servera closed.

11. [student@workstation ~]\$ **ssh student@servera**

12. \_\_\_\_\_

13. | \_ \ | | | | | | | | |\_ \_| ( ) ( )

14. | | / \_ \_ | | | | | | \_ \_ | | | | \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_ \_

15. | // \_ \ \_ | | \_ | / \_ | | | | ' \_ / \_ | | ' \_ \ | | ' \_ \ / \_ |

16. | | \ \ \_ / ( | | | | | | ( | | | | | | | ( | | | | | | | | | ( | |

17. \ | \ \ \_ | \ , \_ | \ | | / \ , \_ | \ | \ / \_ | \ , \_ | | | | | | \ , \_ |

18. \_ / |

19. | \_ /

20.

21. Activate the web console with: **systemctl enable --now cockpit.socket**

22.

23. Register this system with Red Hat Insights: **insights-client --register**

24. Create an account or view all your systems at <https://red.ht/insights-dashboard>

25. Last login: Wed Mar 23 09:21:26 2022 from 172.25.250.9

[student@servera ~]\$

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

27. [student@servera ~]\$ **exit**

28. logout

29. 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 software-rpm**

This concludes the section.

[Previous](#) [Next](#)

# Install and Update Software Packages with DNF

## Objectives

Find, install, and update software packages with the `dnf` command.

## Manage Software Packages with DNF

DNF (Dandified YUM) replaced YUM as the package manager in Red Hat Enterprise Linux 9. DNF commands are functionally the same as YUM commands. For compatibility, YUM commands still exist as symbolic links to DNF:

```
[user@host ~]$ ls -l /bin/ | grep yum | awk '{print $9 " " $10 " " $11}'
yum -> dnf-3
yum-builddep -> /usr/libexec/dnf-utils
yum-config-manager -> /usr/libexec/dnf-utils
yum-debug-dump -> /usr/libexec/dnf-utils
yum-debug-restore -> /usr/libexec/dnf-utils
yumdownloader -> /usr/libexec/dnf-utils
yum-groups-manager -> /usr/libexec/dnf-utils
```

In this course, you work with the `dnf` command. Some documentation might still refer to the `yum` command, but the files are the same linked command.

The low-level `rpm` command can be used to install packages, but it is not designed to work with package repositories or to resolve dependencies from multiple sources automatically.

DNF improves RPM-based software installation and updates. With the `dnf` command, you can install, update, remove, and get information about software packages and their dependencies. You can get a history of transactions and work with multiple Red Hat and third-party software repositories.

## Find Software with DNF

The `dnf help` command displays usage information. The `dnf list` command displays installed and available packages.

```
[user@host ~]$ dnf list 'http*'
```

#### Available Packages

http-parser.i686	2.9.4-6.el9	rhel-9.0-for-x86_64-appstream-rpms
http-parser.x86_64	2.9.4-6.el9	rhel-9.0-for-x86_64-appstream-rpms
httpcomponents-client.noarch	4.5.13-2.el9	rhel-9.0-for-x86_64-appstream-rpms
httpcomponents-core.noarch	4.4.13-6.el9	rhel-9.0-for-x86_64-appstream-rpms
httpd.x86_64	2.4.51-5.el9	rhel-9.0-for-x86_64-appstream-rpms
httpd-devel.x86_64	2.4.51-5.el9	rhel-9.0-for-x86_64-appstream-rpms
httpd-filesystem.noarch	2.4.51-5.el9	rhel-9.0-for-x86_64-appstream-rpms
httpd-manual.noarch	2.4.51-5.el9	rhel-9.0-for-x86_64-appstream-rpms
httpd-tools.x86_64	2.4.51-5.el9	rhel-9.0-for-x86_64-appstream-rpms

The `dnf search KEYWORD` command lists packages by keywords that are in the name and summary fields only. To search for packages with "web server" in their name, summary, and description fields, use `search all`:

```
[user@host ~]$ dnf search all 'web server'
```

```
===== Summary & Description Matched: web server =====
```

```
nginx.x86_64 : A high performance web server and reverse proxy server
```

```
pcp-pmda-weblog.x86_64 : Performance Co-Pilot (PCP) metrics from web server logs
```

```
===== Summary Matched: web server =====
```

```
libcurl.x86_64 : A library for getting files from web servers
```

```
libcurl.i686 : A library for getting files from web servers
```

```
===== Description Matched: web server =====
```

```
freeradius.x86_64 : High-performance and highly configurable free RADIUS server
```

```
git-instaweb.noarch : Repository browser in gitweb
```

```
http-parser.i686 : HTTP request/response parser for C
```

```
http-parser.x86_64 : HTTP request/response parser for C
```

```
httpd.x86_64 : Apache HTTP Server
```

```
mod_auth_openidc.x86_64 : OpenID Connect auth module for Apache HTTP Server
```

```
mod_jk.x86_64 : Tomcat mod_jk connector for Apache
```

```
mod_security.x86_64 : Security module for the Apache HTTP Server
```

```
varnish.i686 : High-performance HTTP accelerator
```

```
varnish.x86_64 : High-performance HTTP accelerator
```

...output omitted...

The `dnf info PACKAGENAME` command returns detailed information about a package, including the needed disk space for installation. For example, the following command retrieves information about the `httpd` package:

```
[user@host ~]$ dnf info httpd

Available Packages
Name           : httpd
Version        : 2.4.51
Release        : 5.el9
Architecture   : x86_64
Size           : 1.5 M
Source         : httpd-2.4.51-5.el9.src.rpm
Repository     : rhel-9.0-for-x86_64-appstream-rpms
Summary        : Apache HTTP Server
URL            : https://httpd.apache.org/
License        : ASL 2.0
Description    : The Apache HTTP Server is a powerful, efficient, and extensible
                  : web server.
```

The `dnf provides PATHNAME` command displays packages that match the specified path name (the path names often include wildcard characters). For example, the following command finds packages that provide the `/var/www/html` directory:

```
[user@host ~]$ dnf provides /var/www/html

httpd-filesystem-2.4.51-5.el9.noarch : The basic directory layout for the Apache HTTP
Server
Repo           : rhel-9.0-for-x86_64-appstream-rpms
Matched from:
Filename       : /var/www/html
```

## Install and Remove Software with DNF

The `dnf install PACKAGENAME` command obtains and installs a software package, including any dependencies.



Dependencies resolved.

## Installing:

Installing dependencies:

### Installing weak dependencies:

## Transaction Summary

Total download size: 2.1 M

Is this ok [y/N]: y

```
(1/10): apr-1.7.0-11.el9.x86_64.rpm      6.4 MB/s | 127 kB    00:00
(2/10): apr-util-bdb-1.6.1-20.el9.x86_64.rpm 625 kB/s | 15 kB     00:00
(3/10): apr-util-openssl-1.6.1-20.el9.x86_64.rpm 1.9 MB/s | 17 kB     00:00
```

Total	24 MB/s	2.1 MB	00:00
-------	---------	--------	-------

```

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : apr-1.7.0-11.el9.x86_64        1/10
  Installing     : apr-util-bdb-1.6.1-20.el9.x86_64 2/10
  Installing     : apr-util-openssl-1.6.1-20.el9.x86_64 3/10
...output omitted...
Installed:
  apr-1.7.0-11.el9.x86_64        apr-util-1.6.1-20.el9.x86_64
  apr-util-bdb-1.6.1-20.el9.x86_64  apr-util-openssl-1.6.1-20.el9.x86_64
...output omitted...
Complete!

```

The `dnf update PACKAGENAME` command obtains and installs a later version of the specified package, including any dependencies. Generally, the process tries to preserve configuration files in place, but in some cases, those files might be renamed if the packager considers that the earlier name will not work after the update. If no `PACKAGENAME` is specified, then it installs all relevant updates.

```
[root@host ~]# dnf update
```

Because a new kernel can be tested only by booting to that kernel, the package specifically supports the installation of multiple versions at once. If the new kernel fails to boot, then the earlier kernel is still available. Running the `dnf update kernel` command installs the new kernel. The configuration files hold a list of packages to always install even if the administrator requests an update.

Use the `dnf list kernel` command to list all installed and available kernels. To view the currently running kernel, use the `uname` command. The `uname` command - `r` option shows only the kernel version and release, and the `uname` command - `a` option shows the kernel release and additional information.

```
[user@host ~]$ dnf list kernel
Installed Packages
```

```
kernel.x86_64                    5.14.0-70.el9                    @System
[user@host ~]$ uname -r
5.14.0-70.el9.x86_64
[user@host ~]$ uname -a
Linux workstation.lab.example.com 5.14.0-70.el9.x86_64 #1 SMP PREEMPT Thu Feb 24 19:11:22 EST 2022 x86_64 x86_64 x86_64 GNU/Linux
```

The `dnf remove PACKAGENAME` command removes an installed software package, including any supported packages.

```
[root@host ~]# dnf remove httpd
```

## Warning

The `dnf remove` command removes the listed packages *and any package that requires the packages to be removed* (and packages which require those packages, and so on). This command can lead to unexpected removal of packages, so review the list of packages to be removed.

## Install and Remove Groups of Software with DNF

The `dnf` command also has the concept of *groups*, which are collections of related software that are installed together.

In Red Hat Enterprise Linux 9, the `dnf` command can install two kinds of package groups. Regular groups are collections of packages. Environment groups are collections of regular groups. The packages or groups that these collections provide might be listed as *mandatory* (they must be installed if the group is installed), *default* (normally installed if the group is installed), or *optional* (not installed when the group is installed, unless specifically requested).

Similar to the `dnf list` command, the `dnf group list` command shows the names of installed and available groups.

```
[user@host ~]$ dnf group list
Available Environment Groups:
    Server with GUI
    Server
    Minimal Install
```

```
...output omitted...
```

Available Groups:

- Legacy UNIX Compatibility

- Console Internet Tools

- Container Management

```
...output omitted...
```

Some groups are normally installed through environment groups and are hidden by default. List these hidden groups with the `dnf group list hidden` command.

The `dnf group info` command displays information about a group. It includes a list of mandatory, default, and optional package names.

```
[user@host ~]$ dnf group info "RPM Development Tools"
```

Group: RPM Development Tools

Description: Tools used for building RPMs, such as `rpmbuild`.

Mandatory Packages:

- `redhat-rpm-config`

- `rpm-build`

Default Packages:

- `rpmdevtools`

Optional Packages:

- `rpmlint`

The `dnf group install` command installs a group that installs its mandatory and default packages and their dependent packages.

```
[root@host ~]# dnf group install "RPM Development Tools"
```

```
...output omitted...
```

Installing Groups:

- RPM Development Tools

Transaction Summary

=====

Install 19 Packages

```
Total download size: 4.7 M
```

```
Installed size: 15 M
```

```
Is this ok [y/N]: y
```

```
...output omitted...
```

## Important

Starting in Red Hat Enterprise Linux 7, the behavior of Yum groups changed, to be treated as objects and tracked by the system. If an installed group is updated, and if the Yum repository added new mandatory or default packages to the group, then those new packages are installed at update.

RHEL 6 and earlier versions consider a group to be installed if all its mandatory packages are installed, or if it had no mandatory packages, or if any default or optional packages in the group are installed. Starting in RHEL 7, a group is considered to be installed *only* if `yum group install` was used to install it. You can use the `yum group mark install GROUPNAME` command to mark a group as installed, and any missing packages and their dependencies are installed at the next update.

RHEL 6 and earlier versions did not have the two-word form of the `yum group` commands. In other words, in RHEL 6 the command `yum grouplist` existed, but the equivalent RHEL 7 and RHEL 8 `yum group list` command did not.

## View Transaction History

All installation and removal transactions are logged in the `/var/log/dnf.rpm.log` file.

```
[user@host ~]$ tail -5 /var/log/dnf.rpm.log
```

```
2022-03-23T16:46:43-0400 SUBDEBUG Installed: python-srpm-macros-3.9-52.el9.noarch
```

```
2022-03-23T16:46:43-0400 SUBDEBUG Installed: redhat-rpm-config-194-1.el9.noarch
```

```
2022-03-23T16:46:44-0400 SUBDEBUG Installed: elfutils-0.186-1.el9.x86_64
```

```
2022-03-23T16:46:44-0400 SUBDEBUG Installed: rpm-build-4.16.1.3-11.el9.x86_64
```

```
2022-03-23T16:46:44-0400 SUBDEBUG Installed: rpmdevtools-9.5-1.el9.noarch
```

The `dnf history` command displays a summary of installation and removal transactions.

```
[root@host ~]# dnf history
```

ID	Command line	Date and time	Action(s)	Altered
7	group install RPM Develop	2022-03-23 16:46	Install	20
6	install httpd	2022-03-23 16:21	Install	10 EE
5	history undo 4	2022-03-23 15:04	Removed	20
4	group install RPM Develop	2022-03-23 15:03	Install	20
3		2022-03-04 03:36	Install	5
2		2022-03-04 03:33	Install	767 EE
1	-y install patch ansible-	2022-03-04 03:31	Install	80

The `dnf history undo` command reverses a transaction.

```
[root@host ~]# dnf history undo 6
...output omitted...
Removing:
  apr-util-openssl x86_64 1.6.1-20.el9 @rhel-9.0-for-x86_64-appstream-rpms 24 k
  httpd            x86_64 2.4.51-5.el9 @rhel-9.0-for-x86_64-appstream-rpms 4.7 M
...output omitted...
```

## Summary of DNF Commands

Packages can be located, installed, updated, and removed by name or by package groups.

Task:	Command:
List installed and available packages by name.	<code>dnf list [NAME-PATTERN]</code>
List installed and available groups.	<code>dnf group list</code>
Search for a package by keyword.	<code>dnf search KEYWORD</code>
Show details of a package.	<code>dnf info PACKAGENAME</code>
Install a package.	<code>dnf install PACKAGENAME</code>
Install a package group.	<code>dnf group install GROUPNAME</code>
Update all packages.	<code>dnf update</code>
Remove a package.	<code>dnf remove PACKAGENAME</code>
Display transaction history.	<code>dnf history</code>

## Manage Package Module Streams with DNF

Traditionally, managing alternative versions of an application's software package and its related packages meant maintaining different repositories for each version. For developers who wanted the latest version of an application and administrators who wanted the most stable version of the application, the resulting situation was tedious to manage. Red Hat simplifies this process by using a technology called *modularity*. With modularity, a single repository can host multiple versions of an application's package and its dependencies.

### Introduction to BaseOS and Application Stream

Red Hat Enterprise Linux 9 distributes the content through two main software repositories: *BaseOS* and *Application Stream* (AppStream).

The BaseOS repository provides the core operating system content for Red Hat Enterprise Linux as RPM packages. BaseOS components have the same lifecycle as content in previous Red Hat Enterprise Linux releases. The Application Stream repository provides content with varying lifecycles as both modules and traditional packages.

Application Stream contains necessary parts of the system, as well as a wide range of applications that were previously available as part of Red Hat Software Collections and other products and programs. Each Application Stream has a lifecycle that is either the same as Red Hat Enterprise Linux 9 or shorter.

Both BaseOS and AppStream are necessary parts of a Red Hat Enterprise Linux 9 system.

The Application Stream repository contains two types of content: modules and traditional RPM packages. A module describes a set of RPM packages that belong together. Modules can contain several streams to make multiple versions of applications available for installation. Enabling a module stream gives the system access to the RPM packages within that module stream. Typically, modules organize the RPM packages around a specific version of a software application or programming language. A typical module contains packages with an application, packages with the application's specific dependency libraries, packages with documentation for the application, and packages with helper utilities.

### Important

Red Hat Enterprise Linux 9.0 ships without modules. Future versions of RHEL 9 might introduce additional content and later software versions as modules. Furthermore, starting with RHEL 9, you must manually specify default module streams, because they are no longer defined by default. You can define default module streams with configuration files in the `/etc/dnf/modules.defaults.d/` directory.

## Module Streams

Each module has one or more *module streams*, which hold different versions of the content. Each of the streams receives updates independently. Think of the module stream as a virtual repository in the Application Stream physical repository.

For each module, you can enable only one of its streams, and this stream provides its packages.

## Module Profiles

Each module can have one or more profiles. A profile is a list of packages that you can install together for a particular use case, such as for a server, client, development, minimal installation, or other.

Installing a module profile installs a particular set of packages from the module stream. You can subsequently install or uninstall packages normally. If you do not specify a profile, then the module installs its default profile.

## Manage Modules with DNF

Red Hat Enterprise Linux 9 supports modular features of Application Stream. To handle the modular content, you can use the `dnf module` command. Otherwise, the `dnf` command works with similar modules to regular packages.

See the following list for some important commands to manage modules:

- **`dnf module list`** : List the available modules with the module name, stream, profiles, and a summary.
- **`dnf module list module-name`** : List the module streams for a specific module and retrieve their status.
- **`dnf module info module-name`** : Display details of a module, including the available profiles and a list of the packages that the module installs. Running the `dnf module info` command without specifying a module stream lists the packages



that are installed from the default profile and stream. Use the *module-name:stream* format to view a specific module stream. Add the `--profile` option to display information about packages that each of the module's profiles installed.

- **dnf module provides *package*** : Display which module provides a specific package.

## References

`dnf(1)` and `dnf.conf(5)` man pages

For more information, refer to the *Managing Software Packages* chapter in the *Red Hat Enterprise Linux 9 Configuring Basic system Settings Guide* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/9/html-single/configuring\\_basic\\_system\\_settings/index#managing-software-packages\\_configuring-basic-system-settings](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/configuring_basic_system_settings/index#managing-software-packages_configuring-basic-system-settings)

For more information, refer to the *Distribution of Content in RHEL 9* chapter in the *Red Hat Enterprise Linux 9 Managing Software with the DNF Tool Guide* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/9/html-single/managing\\_software\\_with\\_the\\_dnf\\_tool/index#assembly\\_distribution-of-content-in-rhel-9\\_managing-software-with-the-dnf-tool](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/managing_software_with_the_dnf_tool/index#assembly_distribution-of-content-in-rhel-9_managing-software-with-the-dnf-tool)

[Modularity](#)

[Previous](#) [Next](#)

## Guided Exercise: Install and Update Software Packages with DNF

In this exercise, you install and remove packages and package groups.

### Outcomes

- Install and remove packages with dependencies.

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 software-dnf
```

## Instructions

1. From workstation, open an SSH session to the servera machine as the student user. Use the `sudo -i` command to switch to the root user.

```
2. [student@workstation ~]$ ssh student@servera
3. ...output omitted...
4. [student@servera ~]$ sudo -i
5. [sudo] password for student: student
```

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

6. Search for a specific package.
  1. Try to run the `nmap` command. You should find that it is not installed.

```
2. [root@servera ~]# nmap
```

```
-bash: nmap: command not found
```

3. Use the `dnf search` command to search for packages with `nmap` as part of their name or summary.

```
4. [root@servera ~]# dnf search nmap
5. ...output omitted...
6. ===== Name Exactly Matched: nmap =====
   =====
7. nmap.x86_64 : Network exploration tool and security scanner
8. ===== Name & Summary Matched: nmap =====
   =====
```

```
nmap-ncat.x86_64 : Nmap's Netcat replacement
```

9. Use the `dnf info` command to obtain more information about the `nmap` package.

```
10. [root@servera ~]# dnf info nmap
```

```
11. ...output omitted...
```

```
12. Available Packages
```

```
13. Name           : nmap
```

```
14. Epoch          : 3
```

```
15. Version        : 7.91
```

```
16. Release        : 10.el9
```

```
...output omitted...
```

7. Use the `dnf install` command to install the `nmap` package.

```
8. [root@servera ~]# dnf install nmap
```

```
9. ...output omitted...
```

```
10. Dependencies resolved.
```

```
11. =====
```

```
12. Package
```

```
13.      Arch      Version           Repository           Size
```

```
14. =====
```

```
15. Installing:
```

```
16. nmap   x86_64   3:7.91-10.el9      rhel-9.0-for-x86_64-appstream-rpms  5.6 M
```

```
17.
```

```
18. Transaction Summary
```

```
19. =====
```

```
20. Install 1 Package
```

```
21.
```

```
22. Total download size: 5.6 M
```

```
23. Installed size: 24 M
```

```
24. Is this ok [y/N]: y
```

```
25. ...output omitted...
```

```
Complete!
```

26. Remove packages.

1. Use the `dnf remove` command to remove the `nmap` package, but respond with `no` when prompted. How many packages are removed?

```

2. [root@servera ~]# dnf remove nmap
3. Dependencies resolved.
4. =====
5. Package
6.      Arch      Version      Repository
7.      Size
8. =====
9. Removing:
10.
11. Transaction Summary
12. =====
13. Remove 1 Package
14.
15. Freed space: 24 M
16. Is this ok [y/N]: n

```

Operation aborted.

17. Use the `dnf remove` command to remove the `tar` package, but respond with `no` when prompted. How many packages are removed?

```

18. [root@servera ~]# dnf remove tar
19. ...output omitted...
20. Dependencies resolved.
21. =====
22. Package      Arch      Version      Repository
23.      Size
24. =====
25. Removing:
26.
27. Transaction Summary
28. =====
29. Remove 1 Package
30.
31. Freed space: 0 M
32. Is this ok [y/N]: n

```

```

26. Removing dependent packages:
27. cockpit          x86_64 264-1.el9      @rhel-9.1-for-x86_64-baseos-rpms
    57 k
28. cockpit-system  noarch 264-1.el9      @System                      3
    .3 M
29. ....output omitted...
30.
31. Transaction Summary
32. =====
    =====
33. Remove    12 Packages
34.
35. Freed space: 48 M
36. Is this ok [y/N]: n

```

Operation aborted.

27. Gather information about the "Security Tools" component group and install it on servera.

1. Use the `dnf group list` command to list all available component groups.

```
[root@servera ~]# dnf group list
```

2. Use the `dnf group info` command to obtain more information about the Security Tools component group, including a list of included packages.

```

3. [root@servera ~]# dnf group info "Security Tools"
4. ...output omitted...
5. Group: Security Tools
6. Description: Security tools for integrity and trust verification.
7. Default Packages:
8.     scap-security-guide
9. Optional Packages:
10.    aide
11.    hmaccalc
12.    openscap
13.    openscap-engine-sce

```

14. openscap-utils
15. scap-security-guide-doc
16. scap-workbench
17. tpm2-tools
18. tss2

udica

19. Use the `dnf group install` command to install the Security Tools component group.

```
20. [root@servera ~]# dnf group install "Security Tools"
21. ...output omitted...
22. Dependencies resolved.
23. =====
    =====
24. Package            Arch  Version            Repository
    Size
25. =====
    =====
26. Installing group/module packages:
27. scap-security-guide
28.                noarch 0.1.60-5.el9  rhel-9.0-for-x86_64-appstream-rpms 6
    83 k
29. Installing dependencies:
30. openscap          x86_64 1:1.3.6-3.el9  rhel-9.0-for-x86_64-appstream-rpms 2
    .0 M
31. ...output omitted...
32.
33. Transaction Summary
34. =====
    =====
35. Install 5 Packages
36.
37. Total download size: 3.0 M
38. Installed size: 94 M
39. Is this ok [y/N]: y
40. ...output omitted...
```

```

41. Installed:
42.  openscap-1:1.3.6-3.el9.x86_64
43.  openscap-scanner-1:1.3.6-3.el9.x86_64
44.  scap-security-guide-0.1.60-5.el9.noarch
45.  xmlsec1-1.2.29-9.el9.x86_64
46.  xmlsec1-openssl-1.2.29-9.el9.x86_64
47.

```

Complete!

## 28. Explore the history and undo options of the dnf command.

1. Use the `dnf history` command to display recent dnf history.

```

2. [root@servera ~]# dnf history
3. ID      | Command line          | Date and time    | Action(s)      | Al
4. -----|-----
5.      3 | group install Security T | 2022-03-24 15:23 | Install        |
6.      2 | install nmap           | 2022-03-24 15:12 | Install        |
7.      1 |
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56. EE

```

On your system, the history is probably different.

7. Use the `dnf history info` command to confirm that the last transaction is the group installation. In the following command, replace the transaction ID with the one from the preceding step.

```

8. [root@servera ~]# dnf history info 3
9. Transaction ID : 3
10. Begin time      : Thu 24 Mar 2022 03:23:56 PM EDT
11. Begin rpmdb     : 7743aed72ac79f632442c9028aafd2499a1591f92a660b3f09219b422
12. End time        : Thu 24 Mar 2022 03:23:58 PM EDT (2 seconds)
13. End rpmdb       : 20c4f0215388b7dca9a874260784b1e5cf9bc142da869967269e3d84d
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.

```

```
14. User           : Student User <student>
15. Return-Code    : Success
16. Releasever     : 9
17. Command Line   : group install Security Tools
18. Comment        :
19. Packages Altered:
20.   Install openscap-1:1.3.6-3.el9.x86_64 @rhel-9.0-for-x86_64-a
    ppstream-rpms
21.   Install openscap-scanner-1:1.3.6-3.el9.x86_64 @rhel-9.0-for-x86_64-a
    ppstream-rpms
```

*...output omitted...*

22. Use the `dnf history undo` command to remove the set of packages that were installed when the `nmap` package was installed. On your system, find the correct transaction ID from the output of the `dnf history` command, and then use that ID in the following command.

```
[root@servera ~]# dnf history undo 2
```

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

```
30. [root@servera ~]# exit
31. logout
32. [student@servera ~]$ exit
33. 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 software-dnf
```

This concludes the section.



# Enable DNF Software Repositories

## Objectives

Enable and disable server use of Red Hat or third-party DNF repositories.

## Enable Red Hat Software Repositories

Systems often have access to many Red Hat repositories. The `dnf repolist all` command lists all available repositories and their statuses:

```
[user@host ~]$ dnf repolist all
```

repo id	repo name	status
rhel-9.0-for-x86_64-appstream-rpms	RHEL 9.0 AppStream	enabled
rhel-9.0-for-x86_64-baseos-rpms	RHEL 9.0 BaseOS	enabled

## Note

Red Hat subscriptions grant access to specific repositories. In the past, administrators needed to attach subscriptions on a per-system basis. Simple Content Access (SCA) simplifies how systems access repositories. With SCA, systems can access any repository from any subscription that you buy, without attaching a subscription. You can enable SCA on the Red Hat Customer Portal within **My Subscriptions** → **Subscription Allocations**, or on your Red Hat Satellite server.

The `dnf config-manager` command can enable and disable repositories. For example, the following command enables the `rhel-9-server-debug-rpms` repository:

```
[user@host ~]$ dnf config-manager --enable rhel-9-server-debug-rpms
```

Non-Red Hat sources provide software through third-party repositories. For example, Adobe provides some of its software for Linux through DNF repositories. In a Red Hat classroom, the `content.example.com` server hosts DNF repositories. The `dnf` command can access repositories from a website, an FTP server, or the local file system.

You can add a third-party repository in one of two ways. You can either create a `.repo` file in the `/etc/yum.repos.d/` directory, or you can add a `[repository]` section

to the `/etc/dnf/dnf.conf` file. Red Hat recommends using `.repo` files, and reserving the `dnf.conf` file for additional repository configurations. The `dnf` command searches both locations by default; however, the `.repo` files take precedence. A `.repo` file contains the URL of the repository, a name, whether to use GPG to verify the package signatures, and if so for the latter, the URL to point to the trusted GPG key.

## Add DNF Repositories

The `dnf config-manager` command can also add repositories to the machine. The following command creates a `.repo` file by using an existing repository's URL.

```
[user@host ~]$ dnf config-manager \
--add-repo="https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/"
Adding repo from: https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/
```

The corresponding `.repo` file is visible in the `/etc/yum.repos.d/` directory:

```
[user@host ~]$ cd /etc/yum.repos.d
[user@host yum.repos.d]$ cat \
dl.fedoraproject.org_pub_epel_9_Everything_x86_64_.repo
[dl.fedoraproject.org_pub_epel_9_Everything_x86_64_]
name=created by dnf config-manager from https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/
baseurl=https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/
enabled=1
```

The `rpm` command uses GPG keys to sign packages, and imports public keys to verify the integrity and authenticity of packages. The `dnf` command uses repository configuration files to provide the GPG public key locations, and imports the keys to verify the packages. Keys are stored in various locations on the remote repository site, such as `http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-9`. Administrators should download the key to a local file rather than for the `dnf` command to retrieve the key from an external source. For example, the following `.repo` file uses the `gpgkey` parameter to reference a local key:

```
[EPEL]
name=EPEL 9
```

```
baseurl=https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-9
```

## RPM Configuration Packages for Local Repositories

Some repositories provide a configuration file and a GPG public key as part of an RPM package to simplify their installation. You can import the GPG public key by using the `rpm --import` command. The `dnf install` command can download and install these RPM packages.

For example, the following command imports the `RPM-GPG-KEY-EPEL-9` (EPEL) GPG public key and installs the RHEL9 Extra Packages for Enterprise Linux (EPEL) repository RPM:

```
[user@host ~]$ rpm --import \
https://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-9
[user@host ~]$ dnf install \
https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

## Warning

Import the RPM GPG key before installing signed packages, to ensure that packages come from a trusted source. If the RPM GPG key is not imported, then the `dnf` command fails to install signed packages.

The `dnf` command `--nogpgcheck` option ignores missing GPG keys, but might result in installing compromised or forged packages.

The `.repo` files often list multiple repository references in a single file. Each repository reference begins with a single-word name in square brackets.

```
[user@host ~]$ cat /etc/yum.repos.d/epel.repo
[epel]
name=Extra Packages for Enterprise Linux $releasever - $basearch
#baseurl=https://download.example/pub/epel/$releasever/Everything/$basearch/
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-$releasever&arch=$basearch&infra=$infra&content=$contentdir
```

```
enabled=1
gpgcheck=1
countme=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-$releasever
...output omitted...
[epel-source]
name=Extra Packages for Enterprise Linux $releasever - $basearch - Source
#baseurl=https://download.example/pub/epel/$releasever/Everything/source/tree/
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-source-$releasever&arch=$basearch&infra=$infra&content=$contentdir
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-$releasever
gpgcheck=1
```

To define a repository, but not to search it by default, insert the `enabled=0` parameter. Although the `dnf config-manager` command persistently enables and disables repositories, the `dnf` command `--enablerepo=PATTERN` and `--disablerepo=PATTERN` options enable and disable repositories temporarily while the command runs.

## References

`dnf(8)`, `dnf.conf(5)`, and `dnf-config-manager(8)` man pages

For more information, refer to the *Managing Software with the DNF Tool* chapter in the Red Hat Enterprise Linux 9 product documentation at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/9/html-single/managing\\_software\\_with\\_the\\_dnf\\_tool](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/managing_software_with_the_dnf_tool)

[Previous](#) [Next](#)

## Guided Exercise: Enable DNF Software Repositories

In this exercise, you configure your server to get packages from a remote DNF repository, and then update or install a package from that repository.

## Outcomes

- Configure a system to obtain software updates from a classroom server, and update the system to use the latest packages.

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 software-repo
```

## Instructions

1. Use the `ssh` command to log in to the `servera` system as the `student` user. Use the `sudo -i` command to switch to the root user.

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

```
3. ...output omitted...
```

```
4. [student@servera ~]$ sudo -i
```

```
5. [sudo] password for student: student
```

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

6. Configure the software repositories on `servera` to obtain custom packages and updates from the following URL:

- Custom packages at `http://content.example.com/rhel9.0/x86_64/rhcsa-practice/rht`
- Updates of the custom packages at `http://content.example.com/rhel9.0/x86_64/rhcsa-practice/errata`
- Use the `dnf config-manager` command to add the custom packages repository.

```
• [root@servera ~]# dnf config-manager \
```

```
• --add-repo "http://content.example.com/rhel9.0/x86_64/rhcsa-practice/rht"
```

```
Adding repo from: http://content.example.com/rhel9.0/x86_64/rhcsa-practice/rht
```

- Examine the software repository file that the previous command created in the `/etc/yum.repos.d` directory. Use the `vim` command to edit the file, and add the `gpgcheck=0` parameter to disable the GPG key check for the repository.

```
[root@servera ~]# vim \
/etc/yum.repos.d/content.example.com_rhel9.0_x86_64_rhcsa-practice_rht.repo
[content.example.com_rhel9.0_x86_64_rhcsa-practice_rht]
name=created by dnf config-manager from http://content.example.com/rhel9.0
/x86_64/rhcsa-practice/rht
baseurl=http://content.example.com/rhel9.0/x86_64/rhcsa-practice/rht
enabled=1
```

```
gpgcheck=0
```

- Create the `/etc/yum.repos.d/errata.repo` file to enable the updates repository with the following content:

```
[rht-updates]
name=rht updates
baseurl=http://content.example.com/rhel9.0/x86_64/rhcsa-practice/errata
enabled=1
```

```
gpgcheck=0
```

- Use the `dnf repolist all` command to list all repositories on the system.

```
[root@servera ~]# dnf repolist all
repo id                                repo name                                status
content.example.com_rhel9.0_x86_64_rhcsa-practice_rht  created by .... enabled
...output omitted...
```

```
rht-updates                            rht updates                            enabled
```

7. Disable the `rht-updates` software repository and install the `rht-system` package.

- Use the `dnf config-manager --disable rht-updates` command to disable the rht-updates repository.

```
[root@servera ~]# dnf config-manager --disable rht-updates
```

- List, and then install, the rht-system package.

```
[root@servera ~]# dnf list rht-system
```

Available Packages

```
rht-system.noarch 1.0.0-1 content.example.com_rhel9.0_x86_64_rhcsa-practice_rht
```

```
[root@servera ~]# dnf install rht-system
```

Dependencies resolved.

```
=====
```

Package Size	Arch	Version	Repository
=====			
Installing:			
rht-system 3.7 k	noarch	1.0.0-1	content..._rht
...output omitted...			
Is this ok [y/N]: y			
...output omitted...			
Installed:			
rht-system-1.0.0-1.noarch			

Complete!

- Verify that the rht-system package is installed, and note the version number of the package.

```
[root@servera ~]# dnf list rht-system
```

Installed Packages

```
rht-system.noarch 1.0.0-1 @content.example.com_rhel9.0_x86_64_rhcsa-practice_rht
```

8. Enable the `rht-updates` software repository and update all relevant software packages.

- Use `dnf config-manager --enable rht-updates` to enable the `rht-updates` repository.

```
[root@servera ~]# dnf config-manager --enable rht-updates
```

- Use the `dnf update` command to update all software packages on `servera`.

```
[root@servera ~]# dnf update
Dependencies resolved.
=====
Package Arch Version Repository
Size
=====
Upgrading:
rht-system noarch 1.0.0-2 rht-updates
7.5 k
...output omitted...
Is this ok [y/N]: y
...output omitted...
```

Complete!

- Verify that the `rht-system` package is upgraded, and note the version number of the package.

```
[root@servera ~]# dnf list rht-system
Installed Packages
```

```
rht-system.noarch 1.0.0-2 @rht-updates
```

9. Exit from `servera`.

```
10. [root@servera ~]# exit
11. logout
12. [student@servera ~]$ exit
13. logout
```



```
14.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 software-repo
```

This concludes the section.

[Previous](#) [Next](#)

## Summary

- Red Hat Subscription Management provides tools to entitle machines to product subscriptions, get updates to software packages, and track information about support contracts and subscriptions that the systems use.
- Software is provided as RPM packages, to install, upgrade, and uninstall software on the system.
- The `rpm` command can query a local database to provide information about the contents of installed packages and to install downloaded package files.
- The `dnf` utility is a powerful command-line tool to install, update, remove, and query software packages.
- Red Hat Enterprise Linux uses Application Streams to provide a single repository to host multiple versions of an application's packages and its dependencies.

[Previous](#) [Next](#)

# Chapter 13. Access Linux File Systems

## Identify File Systems and Devices

### Quiz: Identify File Systems and Devices

## Mount and Unmount File Systems

### Guided Exercise: Mount and Unmount File Systems

## Locate Files on the System

### Guided Exercise: Locate Files on the System

## Lab: Access Linux File Systems

### Summary

#### Abstract

<b>Goal</b>	Access, inspect, and use existing file systems on storage that is attached to a Linux server.
<b>Objectives</b>	<ul style="list-style-type: none"><li>• Identify a directory in the file-system hierarchy and the device where it is stored.</li><li>• Access the contents of file systems by adding and removing file systems in the file-system hierarchy.</li><li>• Search for files on mounted file systems with the <code>find</code> and <code>locate</code> commands.</li></ul>
<b>Sections</b>	<ul style="list-style-type: none"><li>• Identify File Systems and Devices (and Quiz)</li><li>• Mount and Unmount File Systems (and Guided Exercise)</li><li>• Locate Files on the System (and Guided Exercise)</li></ul>
<b>Lab</b>	Access Linux File Systems

## Identify File Systems and Devices

### Objectives

Identify a directory in the file-system hierarchy and the device where it is stored.

### Storage Management Concepts

Red Hat Enterprise Linux (RHEL) uses the *Extents File System (XFS)* as the default local file system. RHEL supports the *Extended File System (ext4)* file system for managing local files. Starting with RHEL 9, the Extensible File Allocation Table (exFAT) file system is supported for removable media use. In an enterprise server cluster, shared disks use the *Global File System 2 (GFS2)* file system to manage concurrent multi-node access.

## File Systems and Mount Points

Access the contents of a file system by mounting it on an empty directory. This directory is called a mount point. When the directory is mounted, use the `ls` command to list its contents. Many file systems are automatically mounted when the system boots.

A mount point differs slightly from a Microsoft Windows drive letter, where each file system is a separate entity. Mount points enable multiple file system devices to be available in a single tree structure. This mount point is similar to NTFS mounted folders in Microsoft Windows.

## File Systems, Storage, and Block Devices

A block device is a file that provides low-level access to storage devices. A block device must be optionally partitioned, and a file system that was created before the device can be mounted.

The `/dev` directory stores block device files, which RHEL creates automatically for all devices. In RHEL 9, the first detected SATA, SAS, SCSI, or USB hard drive is called the `/dev/sda` device; the second is the `/dev/sdb` device; and so on. These names represent the entire hard drive.

**Table 13.1. Block Device Naming**

Type of device	Device naming pattern
SATA/SAS/USB-attached storage (SCSI driver)	<code>/dev/sda</code> , <code>/dev/sdb</code> , <code>/dev/sdc</code> , ...
virtio-blk paravirtualized storage (VMs)	<code>/dev/vda</code> , <code>/dev/vdb</code> , <code>/dev/vdc</code> , ...
virtio-scsi paravirtualized storage (VMs)	<code>/dev/sda</code> , <code>/dev/sdb</code> , <code>/dev/sdc</code> , ...
NVMe-attached storage (SSDs)	<code>/dev/nvme0</code> , <code>/dev/nvme1</code> , ...
SD/MMC/eMMC storage (SD cards)	<code>/dev/mmcblk0</code> , <code>/dev/mmcblk1</code> , ...

## Disk Partitions

Usually, the entire storage device is not created into one file system. To create a partition, divide the storage devices into smaller chunks.

With partitions, you can compartmentalize a disk: the various partitions might be formatted with different file systems or be used for other purposes. For example, one partition might contain user home directories, whereas another partition might

contain system data and logs. Even when the home directory partition is loaded with data, the system partition might still have available space.

Partitions are block devices in their own right. For example, on the first SATA-attached storage, the first partition is the `/dev/sda1` disk. The second partition of the same storage is the `/dev/sda2` disk. The third partition on the third SATA-attached storage device is the `/dev/sdc3` disk, and so on. Paravirtualized storage devices have a similar naming system. For example, the first partition on the first storage device is the `/dev/vda1` disk. The second partition of the second storage device is the `/dev/vdb2` disk, and so on.

An NVMe-attached SSD device names its partitions differently from a SATA-attached device. For NVMe storage devices, the `nvme` part of the name refers to the device; the `nv` part refers to the namespace; and the `pz` part refers to the partition. For example, the first partition for the first namespace on the first disk is the `/dev/nvme0n1p1` partition. The third partition for the first namespace on the second disk is the `/dev/nvme1n1p3` partition, and so on.

SD or MMC cards can sometimes have a similar naming system to the SATA devices (`/dev/sdM`). In some cases, SD or MMC cards might have names such as `/dev/mmcblk0p1`, where the `mmcblkx` part of the name refers to the storage device, and the `px` part of the name refers to the partition number on that device.

An extended listing of the `/dev/sda1` device file on the host machine reveals the `b` file type, which stands for a block device:

```
[user@host ~]$ ls -l /dev/sda1  
brw-rw----. 1 root disk 8, 1 Feb 22 08:00 /dev/sda1
```

## Logical Volumes

Another way of organizing disks and partitions is with *Logical Volume Management (LVM)*. With LVM, it is possible to aggregate block devices into a volume group. Disk space in the volume group is separated into logical volumes, which are the functional equivalent of a partition on a physical disk.

The LVM system assigns names to volume groups and logical volumes on their creation. LVM creates a directory in the `/dev` directory that matches the group name, and creates a symbolic link within that new directory with the same name as the logical volume. That logical volume file is then available to be mounted. For

example, when a `myvg` volume group and the `mylv` logical volume are present, the full path to the logical volume is the `/dev/myvg/mylv` file.

## Note

The previously mentioned logical volume device name establishes a symbolic link to the device file that accesses it, which might vary between boots. Another form of logical volume device name, which is linked from files in the `/dev/mapper` directory, is often used for symbolic links to the device file.

## Examine File Systems

Use the `df` command to display an overview of local and remote file-system devices, which includes the total disk space, used disk space, free disk space, and the percentage of the entire disk space.

The following example displays the file systems and mount points on the host machine:

```
[user@host ~]$ df
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
devtmpfs	912584	0	912584	0%	/dev
tmpfs	936516	0	936516	0%	/dev/shm
tmpfs	936516	16812	919704	2%	/run
tmpfs	936516	0	936516	0%	/sys/fs/cgroup
/dev/vda3	8377344	1411332	6966012	17%	/
/dev/vda1	1038336	169896	868440	17%	/boot
tmpfs	187300	0	187300	0%	/run/user/1000

The partitioning shows that two physical file systems are mounted on the `/` and `/boot` directories that commonly exist on virtual machines.

The `tmpfs` and `devtmpfs` devices are file systems in system memory. All files that are written to the `tmpfs` or `devtmpfs` file system disappear after a system reboot.

The `df` command `-h` or `-H` options are human-readable options to improve the readability of the output sizes. The `-h` option reports in KiB ( $2^{10}$ ), MiB ( $2^{20}$ ), or GiB ( $2^{30}$ ), whereas the `-H` option reports in SI units: KB ( $10^3$ ), MB ( $10^6$ ), or GB ( $10^9$ ). Hard drive manufacturers usually use SI units when advertising their products.

View the file systems on the host machine with all units converted to human-readable format:

```
[user@host ~]$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	892M	0	892M	0%	/dev
tmpfs	915M	0	915M	0%	/dev/shm
tmpfs	915M	17M	899M	2%	/run
tmpfs	915M	0	915M	0%	/sys/fs/cgroup
/dev/vda3	8.0G	1.4G	6.7G	17%	/
/dev/vda1	1014M	166M	849M	17%	/boot
tmpfs	183M	0	183M	0%	/run/user/1000

Use the `du` command for more detailed information about a specific directory tree space. The `du` command `-h` and `-H` options convert the output to a human-readable format. The `du` command shows the size of all files in the current directory tree recursively.

View the disk usage report for the `/usr/share` directory on the host machine:

```
[root@host ~]# du /usr/share
...output omitted...
176 /usr/share/smartmontools
184 /usr/share/nano
8 /usr/share/cmake/bash-completion
8 /usr/share/cmake
356676 /usr/share
```

View the disk usage report in human-readable format for the `/usr/share` directory:

```
[root@host ~]# du -h /usr/share
...output omitted...
176K /usr/share/smartmontools
184K /usr/share/nano
8.0K /usr/share/cmake/bash-completion
8.0K /usr/share/cmake
```

## References

df(1) and du(1) man pages

[Next](#)

## Quiz: Identify File Systems and Devices

Choose the correct answers to the following questions:

1.

2.

- 1.** What is the device file name of an entire SATA hard drive in the /dev directory?

- A /dev/vda
- B /dev/sda1
- C /dev/vg\_install/lv\_home
- D /dev/sda

3. CheckResetShow Solution

4.

5.

- 2.** Which command

displays  
the file  
systems  
with the  
mount  
points?

- A `du -H`
- B `df`
- C `du`
- D `ls`

6. CheckResetShow Solution

7.

8.

3. Which command displays the disk usage report in human-readable format for the /home directory?

- A `ls /home`
- B `df`
- C `du -h /home`
- D `du /home`

9. CheckResetShow Solution

10.

11.



4. What is the correct device file name for the third partition on the second virtio-blk disk that is attached to a virtual machine?

- A `/dev/vdb3`
- B `/dev/vda2`
- C `/dev/sda3`
- D `/dev/vda3`

12. CheckResetShow Solution

13.

14.

5. Which command provides an overview of the file-system mount points and the available free space in SI units?

- A `df`
- B `df -h`

C

df -H

D

du -h

15. CheckResetShow Solution

[Previous](#) [Next](#)

## Mount and Unmount File Systems

### Objectives

Access the contents of file systems by adding and removing file systems in the file-system hierarchy.

### Mount File Systems Manually

To access the file system on a removable storage device, you must mount the storage device. With the `mount` command, the `root` user can mount a file system manually. The first argument of the `mount` command specifies the file system to mount. The second argument specifies the directory as the mount point in the file-system hierarchy.

You can mount the file system in one of the following ways with the `mount` command:

- With the device file name in the `/dev` directory.
- With the UUID, a universally unique identifier of the device.

Then, identify the device to mount, ensure that the mount point exists, and mount the device on the mount point.

### Note

If you mount a file system with the `mount` command, and then reboot your system, the file system is not automatically remounted. The *Red Hat System Administration II* (RH134) course explains how to persistently mount file systems with the `/etc/fstab` file.

## Identify a Block Device

A hot-pluggable storage device, whether a hard disk drive (HDD) or a solid-state device (SSD) in a server, or alternatively a USB storage device, might be plugged each time into a different port on a system. Use the `lsblk` command to list the details of a specified block device or of all the available devices.

```
[root@host ~]# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
vda	252:0	0	10G	0	disk	
├─vda1	252:1	0	1M	0	part	
├─vda2	252:2	0	200M	0	part	/boot/efi
├─vda3	252:3	0	500M	0	part	/boot
└─vda4	252:4	0	9.3G	0	part	/
vdb	252:16	0	5G	0	disk	
vdc	252:32	0	5G	0	disk	
vdd	252:48	0	5G	0	disk	

The partition size helps to identify the device when the partition name is unknown. For example, considering the previous output, if the size of the identified partition is 9.3 GB, then mount the `/dev/vda4` partition.

## Mount File System with the Partition Name

The following example mounts the `/dev/vda4` partition on the `/mnt/data` mount point.

```
[root@host ~]# mount /dev/vda4 /mnt/data
```

The mount point directory must exist before mounting the file system. The `/mnt` directory exists for use as a temporary mount point.

## Important

If a directory to use as a mount point is not empty, then the existing files are hidden and not accessible when a file system is mounted there. The original files are accessible again after the mounted file system is unmounted.

Device detection order and storage device naming can change when devices are added or removed on a system. It is recommended to use an unchanging device identifier to mount file systems consistently.

## Mount File System with Partition UUID

One stable identifier that is associated with a file system is its universally unique identifier (UUID). This UUID is stored in the file system superblock and remains the same until the file system is re-created.

The `lsblk -fp` command lists the full path of the device, the UUIDs and mount points, and the partition's file-system type. The mount point is blank when the file system is not mounted.

```
[root@host ~]# lsblk -fp
```

NAME	FSTYPE	FSVER	LABEL	UUID	FSAVAIL	FSUSE%	MOUNTPOINTS
/dev/vda							
└─/dev/vda1							
└─/dev/vda2	vfat	FAT16		7B77-95E7	192.3M	4%	/boot/efi
└─/dev/vda3	xfss		boot	2d67e6d0-...-1f091bf1	334.9M	32%	/boot
└─/dev/vda4	xfss		root	efd314d0-...-ae98f652	7.7G	18%	/
/dev/vdb							
/dev/vdc							
/dev/vdd							

Mount the file system by the file-system UUID.

```
[root@host ~]# mount UUID="efd314d0-b56e-45db-bbb3-3f32ae98f652" /mnt/data
```

## Automatically Mount Removable Storage Devices

With the graphical desktop environment, the system automatically mounts removable storage media when the media presence is detected.

The removable storage device mounts at the `/run/media/USERNAME/LABEL` location. *USERNAME* is the name of the user that is logged in to the graphical environment. *LABEL* is an identifier, which is typically the label on the storage media.

To safely detach a removable device, manually unmount all file systems on the device first.

## Unmount File Systems

System shutdown and reboot procedures unmount all file systems automatically. All file-system data is flushed to the storage device, to ensure file-system data integrity.

### Warning

File-system data uses memory cache during normal operation. You must unmount a removable drive's file systems before unplugging the drive. The unmount procedure flushes data to disk before releasing the drive.

The `umount` command uses the mount point as an argument to unmount a file system.

```
[root@host ~]# umount /mnt/data
```

Unmounting is not possible when the mounted file system is in use. For the `umount` command to succeed, all processes must stop accessing data under the mount point.

In the following example, the `umount` command fails because the shell uses the `/mnt/data` directory as its current working directory, and thus generates an error message.

```
[root@host ~]# cd /mnt/data
[root@host data]# umount /mnt/data
umount: /mnt/data: target is busy.
```

The `lsdf` command lists all open files and the processes that are accessing the file system. The list helps to identify which processes are preventing the file system from successfully unmounting.

```
[root@host data]# lsdf /mnt/data
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
bash	1593	root	cwd	DIR	253,17		6 128	/mnt/data

```
lsuf 2532 root cwd DIR 253,17 19 128 /mnt/data
lsuf 2533 root cwd DIR 253,17 19 128 /mnt/data
```

Identify and wait for the processes to complete, or send the SIGTERM or SIGKILL signal to terminate them. In this case, it is sufficient to change to a current working directory that is outside the mount point.

```
[root@host data]# cd
[root@host ~]# umount /mnt/data
```

## References

lsblk(8), mount(8), umount(8), and lsuf(8) man pages

[Previous](#) [Next](#)

## Guided Exercise: Mount and Unmount File Systems

In this exercise, you practice mounting and unmounting file systems.

### Outcomes

- Identify and mount a new file system at a specified mount point, and then unmount the file 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 fs-mount
```

### Instructions

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

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

3. ...output omitted...
4. [student@servera ~]\$ **sudo -i**
5. [sudo] password for student: **student**

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

6. A new partition with a file system is added to the /dev/vdb disk on the servera machine. Mount the newly available partition by using the UUID at the /mnt/part1 mount point.

1. Create the /mnt/part1 directory.

```
[root@servera ~]# mkdir /mnt/part1
```

2. Query the UUID of the /dev/vdb1 device.

```
3. [root@servera ~]# lsblk -fp /dev/vdb
```

```
4. NAME           FSTYPE LABEL UUID                                MOUNTPOINT
5. /dev/vdb
```

```
└─/dev/vdb1 xfs          a04c511a-b805-4ec2-981f-42d190fc9a65
```

6. Mount the file system by using the UUID on the /mnt/part1 directory. Use the /dev/vdb1 UUID from the previous command output.

```
7. [root@servera ~]# mount \
```

```
UUID="a04c511a-b805-4ec2-981f-42d190fc9a65" /mnt/part1
```

8. Verify that the /dev/vdb1 device is mounted on the /mnt/part1 directory.

```
9. [root@servera ~]# lsblk -fp /dev/vdb
```

```
10. NAME           FSTYPE LABEL UUID                                MOUNTPOINT
11. /dev/vdb
```

```
└─/dev/vdb1 xfs          a04c511a-b805-4ec2-981f-42d190fc9a65 /mnt/part1
```

7. Change to the /mnt/part1 directory and create the testdir subdirectory. Create the /mnt/part1/testdir/newmount file.

1. Change to the /mnt/part1 directory.

```
[root@servera ~]# cd /mnt/part1
```

2. Create the /mnt/part1/testdir directory.

```
[root@servera part1]# mkdir testdir
```

3. Create the /mnt/part1/testdir/newmount file.

```
[root@servera part1]# touch testdir/newmount
```

8. Unmount the file system that is mounted on the /mnt/part1 directory.

1. Unmount the /mnt/part1 directory when the shell is in the /mnt/part1 directory. The umount command fails to unmount the device.

```
2. [root@servera part1]# umount /mnt/part1
```

```
umount: /mnt/part1: target is busy.
```

3. Change the current directory on the shell to the /root directory.

```
4. [root@servera part1]# cd
```

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

5. Unmount the /mnt/part1 directory.

```
[root@servera ~]# umount /mnt/part1
```

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

```
10. [root@servera ~]# exit
```

```
11. logout
```

```
12. [student@servera ~]$ exit
```

```
13. logout
```

```
14. 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 fs-mount
```

This concludes the section.

[Previous](#) [Next](#)

## Locate Files on the System

### Objectives

Search for files on mounted file systems with the `find` and `locate` commands.

### Search for Files

A system administrator needs tools to search for files that match specific criteria on the file system. This section discusses two commands to search for files in the file-system hierarchy:

- The `locate` command searches a pre-generated index for file names or file paths, and returns the results instantly.
- The `find` command searches for files in real time by parsing the file-system hierarchy.

### Locate Files by Name

The `locate` command searches for files based on the name or path to the file. The command is fast, because it looks up this information from the `mlocate` database. However, this database does not update in real time and requires frequent updates for accurate results. This feature also means that the `locate` command does not search for files that were created since the last database update.

The `locate` database updates automatically every day. However, the `root` user might issue the `updatedb` command to force an immediate update.

```
[root@host ~]# updatedb
```

The `locate` command restricts results for unprivileged users. To see the resulting file name, the user must have search permission on the directory where the file resides. For example, locate the files that the `developer` user can read, and that match the `passwd` keyword in the name or path:

```
[developer@host ~]$ locate passwd
/etc/passwd
/etc/passwd-
/etc/pam.d/passwd
...output omitted...
```

The following example shows the file name or path for a partial match with the search query:

```
[root@host ~]# locate image
/etc/selinux/targeted/contexts/virtual\_image_context
/usr/bin/grub2-mkimage
/usr/lib/sysimage
...output omitted...
```

The `locate` command `-i` option performs a case-insensitive search. This option returns all possible combinations of matching uppercase and lowercase letters:

```
[developer@host ~]$ locate -i messages
...output omitted...
/usr/share/locale/zza/LC_MESSAGES
/usr/share/makedumpfile/eppic_scripts/ap_messages_3_10_to_4_8.c
/usr/share/vim/vim82/ftplugin/msmessages.vim
...output omitted...
```

The `locate` command `-n` option limits the number of returned search results. The following example limits the search results from the `locate` command to the first five matches:

```
[developer@host ~]$ locate -n 5 passwd
```

```
/etc/passwd  
/etc/passwd-  
/etc/pam.d/passwd  
...output omitted...
```

## Search for Files in Real Time

The `find` command locates files by searching in real time in the file-system hierarchy. This command is slower but more accurate than the `locate` command. The `find` command also searches for files based on criteria other than the file name, such as the file's permissions, type of file, size, or modification time.

The `find` command looks at files in the file system with the user account that executed the search. The user that runs the `find` command must have read and execute permission on a directory to examine its contents.

The first argument to the `find` command is the directory to search. If the `find` command omits the directory argument, then it starts the search in the current directory and looks for matches in any subdirectory.

To search for files by file name, use the `find` command `-name FILENAME` option to return the path of files that match `FILENAME` exactly. For example, to search for the `sshd_config` files in the root `/` directory, run the following command:

```
[root@host ~]# find / -name sshd_config  
/etc/ssh/sshd_config
```

## Note

In the `find` command, the complete word options use a single dash for options, unlike a double dash for most other Linux commands.

Wildcards are available to search for a file name and to return all results for a partial match. With wildcards, it is essential to quote the file name, to prevent the terminal from misinterpreting the wildcard.

In the following example, starting in the `/` directory, search for files that end with the `.txt` extension:

```
[root@host ~]# find / -name '*.txt'
```

```
...output omitted...  
/usr/share/libgpg-error/errorref.txt  
/usr/share/licenses/audit-libs/lgpl-2.1.txt  
/usr/share/licenses/pam/gpl-2.0.txt  
...output omitted...
```

To search for files in the `/etc/` directory that contain the `pass` string, run the following command:

```
[root@host ~]# find /etc -name '*pass*'  
/etc/passwd-  
/etc/passwd  
/etc/security/opasswd  
...output omitted...
```

To perform a case-insensitive search for a file name, use the `find` command - `iname` option, followed by the file name to search. To search files with case-insensitive text that matches the `messages` string in their names in the root `/` directory, run the following command:

```
[root@host ~]# find / -iname '*messages*'  
/sys/power/pm_debug_messages  
/usr/lib/locale/C.utf8/LC_MESSAGES  
/usr/lib/locale/C.utf8/LC_MESSAGES/SYS_LC_MESSAGES  
...output omitted...
```

## Search for Files Based on Ownership or Permission

The `find` command searches for files based on their ownership or permissions. The `find` command `-user` and `-group` options search by a user and group name, or by user ID and group ID.

To search for files in the `/home/developer` directory that the `developer` user owns:

```
[developer@host ~]$ find -user developer  
.  
./bash_logout
```

```
./bash_profile  
...output omitted...
```

To search for files in the `/home/developer` directory that the `developer` group owns:

```
[developer@host ~]$ find -group developer  
.  
./bash_logout  
./bash_profile  
...output omitted...
```

To search for files in the `/home/developer` directory that the `1000` user ID owns:

```
[developer@host ~]$ find -uid 1000  
.  
./bash_logout  
./bash_profile  
...output omitted...
```

To search for files in the `/home/developer` directory that the `1000` group ID owns:

```
[developer@host ~]$ find -gid 1000  
.  
./bash_logout  
./bash_profile  
...output omitted...
```

The `find` command `-user` and `-group` options search for files where the file owner and group owner are different. The following example lists files that the `root` user owns and with the `mail` group:

```
[root@host ~]# find / -user root -group mail  
/var/spool/mail  
...output omitted...
```

The `find` command `-perm` option looks for files with a particular permission set. The octal values define the permissions with 4, 2, and 1 for read, write, and execute. Permissions are preceded with a `/` or `-` sign to control the search results.

Octal permission that is preceded by the `/` sign matches files where at least one permission is set for user, group, or other for that permission set. A file with the `r--r--r--` permissions does not match the `/222` permission but matches the `rw-r--r--` permission. A `-` sign before the permission means that all three parts of the permissions must match. For the previous example, files with the `rw-rw-rw-` permissions match. You can also use the `find` command `-perm` option with the symbolic method for permissions.

For example, the following commands match any file in the `/home` directory for which the owning user has read, write, and execute permissions, and members of the owning group have read and write permissions, and others have read-only access. Both commands are equivalent; the first one uses the octal method for permissions, whereas the second one uses the symbolic methods.

```
[root@host ~]# find /home -perm 764
...output omitted...
[root@host ~]# find /home -perm u=rwx,g=rw,o=r
...output omitted...
```

The `find` command `-ls` option is convenient when searching files by permissions, because it provides information for the files that includes their permissions.

```
[root@host ~]# find /home -perm 764 -ls
26207447  0 -rwxrw-r--  1 user  user   0 May 10 04:29 /home/user/file1
```

To search for files for which the user has at least write and execute permissions, and the group has at least write permission, and others have at least read permission, run the following command:

```
[root@host ~]# find /home -perm -324
...output omitted...
[root@host ~]# find /home -perm -u=wx,g=w,o=r
...output omitted...
```

To search for files for which the user has read permissions, or the group has at least read permissions, or others have at least write permission, run the following command:

```
[root@host ~]# find /home -perm /442
...output omitted...
[root@host ~]# find /home -perm /u=r,g=r,o=w
...output omitted...
```

When used with / or - signs, the 0 value works as a wildcard, because it means any permission.

To search for any file in the /home/developer directory for which others have at least read access on the host machine, run the following command:

```
[developer@host ~]$ find -perm -004
...output omitted...
[developer@host ~]$ find -perm -o=r
...output omitted...
```

To search for all files in the /home/developer directory where others have write permission, run the following command:

```
[developer@host ~]$ find -perm -002
...output omitted...
[developer@host ~]$ find -perm -o=w
...output omitted...
```

## Find Files Based on Size

The `find` command `-size` option is followed by a numeric value, and the unit looks up files that match a specified size. Use the following list for the units with the `find` command `-size` option:

- For kilobytes, use the `k` unit with `k` always in lowercase.
- For megabytes, use the `M` unit with `M` always in uppercase.
- For gigabytes, use the `G` unit with `G` always in uppercase.

You can use the plus + and minus - characters to include files that are larger and smaller than the given size, respectively. The following example shows a search for files with an exact size of 10 megabytes:

```
[developer@host ~]$ find -size 10M  
...output omitted...
```

To search for files with a size of more than 10 gigabytes:

```
[developer@host ~]$ find -size +10G  
...output omitted...
```

To search for files with a size of less than 10 kilobytes:

```
[developer@host ~]$ find -size -10k  
...output omitted...
```

## Important

The `find` command `-size` option rounds everything to single units. For example, the `find -size 1M` command shows files that are smaller than 1 MB, because it rounds up all files to 1 MB.

### Search for Files Based on Modification Time

The `find` command `-mmin` option, followed by the time in minutes, searches for all files with content that changed `n` minutes ago. The file's time stamp is rounded down and supports fractional values with the `+n` and `-n` range.

To search for all files with content that changed 120 minutes ago:

```
[root@host ~]# find / -mmin 120  
...output omitted...
```

The `+` modifier in front of the minutes finds all files in the `/` directory that changed more than `n` minutes ago. To search for all files with content that changed more than 200 minutes ago:

```
[root@host ~]# find / -mmin +200
```



```
...output omitted...
```

The `-` modifier searches for all files in the `/` directory that changed less than `n` minutes ago. The following example lists files that changed less than 150 minutes ago:

```
[root@host ~]# find / -mmin -150  
...output omitted...
```

## Search for Files Based on File Type

The `find` command `-type` option limits the search scope to a given file type. Use the following flags to limit the search scope:

- For regular files, use the `f` flag.
- For directories, use the `d` flag.
- For soft links, use the `l` flag.
- For block devices, use the `b` flag.

Search for all directories in the `/etc` directory:

```
[root@host ~]# find /etc -type d  
/etc  
/etc/tmpfiles.d  
/etc/systemd  
/etc/systemd/system  
/etc/systemd/system/getty.target.wants  
...output omitted...
```

Search for all soft links in the `/` directory:

```
[root@host ~]# find / -type l  
...output omitted...
```

Search for all block devices in the `/dev` directory:

```
[root@host ~]# find /dev -type b  
/dev/vda1
```

```
/dev/vda
```

The `find` command `-links` option followed by a number looks for all files with a specific hard link count. The number preceded by a `+` modifier looks for files with a higher count than the given hard link count. If the number precedes a `-` modifier, then the search is limited to files with a lower hard link count than the given number.

Search for all regular files with more than one hard link:

```
[root@host ~]# find / -type f -links +1  
...output omitted...
```

## References

`locate(1)`, `updatedb(8)`, and `find(1)` man pages

[Previous](#) [Next](#)

## Guided Exercise: Locate Files on the System

In this exercise, you search for specific files on mounted file systems by using the `find` and `locate` commands.

### Outcomes

- Search for files with the `find` and `locate` commands.

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 fs-locate
```

## Instructions

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

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

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

4. Use the `locate` command to search for files on the servera machine.
  1. Update the `locatedb` database manually on the server machine. Use the `sudo updatedb` command to update the database.

```
2. [student@servera ~]$ sudo updatedb
3. [sudo] password for student: student
```

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

4. Locate the `logrotate.conf` configuration file.

```
5. [student@servera ~]$ locate logrotate.conf
6. /etc/logrotate.conf
```

```
/usr/share/man/man5/logrotate.conf.5.gz
```

7. Locate the `networkmanager.conf` configuration file, ignoring case sensitivity.

```
8. [student@servera ~]$ locate -i networkmanager.conf
9. /etc/NetworkManager/NetworkManager.conf
10. /etc/dbus-1/system.d/org.freedesktop.NetworkManager.conf
```

```
/usr/share/man/man5/NetworkManager.conf.5.gz
```

5. Use the `find` command to search in real time on the servera machine according to the following requirements:
  1. List all files in the `/var/lib` directory that the `chrony` user owns.
  2. List all files in the `/var` directory that the `root` user and the `mail` group own.

3. List all files in the `/usr/bin` directory with a file size that is greater than 50 KB.
4. List all files in the `/home/student` directory that changed in the last 120 minutes.
5. List all the block device files in the `/dev` directory.
6. Search for all files in the `/var/lib` directory that the `chrony` user owns, with root privilege.

```
7. [student@servera ~]$ sudo find /var/lib -user chrony
8. [sudo] password for student: student
9. /var/lib/chrony
```

```
/var/lib/chrony/drift
```

10. List all files in the `/var` directory that the `root` user owns and that belong to the `mail` group.

```
11. [student@servera ~]$ sudo find /var -user root -group mail
```

```
/var/spool/mail
```

12. List all files in the `/usr/bin` directory with a greater file size than 50 KB.

```
13. [student@servera ~]$ find /usr/bin -size +50k
14. /usr/bin/iconv
15. /usr/bin/locale
16. /usr/bin/localedef
17. /usr/bin/cmp
```

```
...output omitted...
```

18. List all files in the `/home/student` directory that changed in the last 120 minutes.

```
19. [student@servera ~]$ find /home/student -mmin -120
20. /home/student/.bash_logout
21. /home/student/.bash_profile
22. /home/student/.bashrc
```

...output omitted...

23. List all block device files in the /dev directory.

```
24. [student@servera ~]$ find /dev -type b
```

```
25. /dev/vdd
```

```
26. /dev/vdc
```

```
27. /dev/vdb
```

```
28. /dev/vda3
```

```
29. /dev/vda2
```

```
30. /dev/vda1
```

```
/dev/vda
```

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

```
7. [student@servera ~]$ exit
```

```
8. logout
```

```
9. 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 fs-locate
```

This concludes the section.

[Previous](#) [Next](#)

## Summary

- Storage devices are represented by the *block device* file type.

- The `df` command reports total disk space, used disk space, and free disk space on all mounted regular file systems.
- The `root` user can use the `mount` command to manually mount a file system.
- To successfully unmount a device, all processes must stop accessing the mount point.
- The removable storage devices are mounted in the `/run/media` directory when using the graphical environment.
- The `lsblk` command lists the details of block devices, such as the size and the UUID.
- The `find` command searches in real time in the local file systems for files according to search criteria.

[Previous](#) [Next](#)

## Chapter 14. Analyze Servers and Get Support

### Analyze and Manage Remote Servers

#### Guided Exercise: Analyze and Manage Remote Servers

#### Create a Diagnostics Report

#### Guided Exercise: Create a Diagnostics Report

#### Detect and Resolve Issues with Red Hat Insights

#### Quiz: Detect and Resolve Issues with Red Hat Insights

### Summary

### **Abstract**

<b>Goal</b>	Investigate and resolve issues in the web-based management interface, getting support from Red Hat to help solve problems.
<b>Objectives</b>	<ul style="list-style-type: none"> <li>• Activate the web console management interface to remotely manage and monitor the performance of a Red Hat Enterprise Linux server.</li> <li>• Describe and use the Red Hat Customer Portal key resources to find information from Red Hat documentation and the Knowledgebase.</li> <li>• Use Red Hat Insights to analyze servers for issues, remediate or resolve them, and confirm that the solution worked.</li> </ul>
<b>Sections</b>	<ul style="list-style-type: none"> <li>• Analyze and Manage Remote Servers (and Guided Exercise)</li> <li>• Create a Diagnostics Report (and Guided Exercise)</li> <li>• Detect and Resolve Issues with Red Hat Insights (and Quiz)</li> </ul>

# Analyze and Manage Remote Servers

## Objectives

Activate the web console management interface to remotely manage and monitor the performance of a Red Hat Enterprise Linux server.

## Describe the Web Console

The *web console* is a web-based management interface for Red Hat Enterprise Linux. The interface is designed for managing and monitoring your servers, and is based on the open-source Cockpit service.

You can use the web console to monitor system logs and to view graphs of system performance. Additionally, you can use your web browser to change settings by using graphical tools in the web console interface, including a fully functional interactive terminal session.

## Enable the Web Console

Starting from Red Hat Enterprise Linux 7, the web console is installed by default in all installation variants except in a minimal installation. You can use the following command to install the web console:

```
[root@host ~]# dnf install cockpit
```

Then, enable and start the `cockpit.socket` service, which runs a web server. This step is necessary if you need to connect to the system through the web interface.

```
[root@host ~]# systemctl enable --now cockpit.socket
```

```
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket -> /usr/lib/systemd/system/cockpit.socket.
```

If you are using a custom firewall profile, then you must add the `cockpit` service to `firewalld` to open port 9090 in the firewall:

```
[root@host ~]# firewall-cmd --add-service=cockpit --permanent
```

```
success
```

```
[root@host ~]# firewall-cmd --reload
```

success

## Log in to the Web Console

The web console provides its own web server. Launch your web browser to log in to the web console.

Open `https://servername:9090` in your web browser, where *servername* is the hostname or IP address of your server. The web console protects the connection by a *Transport Layer Security (TLS)* session. By default, the `cockpit` service installs the web console with a self-signed TLS certificate. When you connect to the web console for the first time, the web browser probably displays a security warning. The `cockpit-ws(8)` man page provides instructions on how to replace the TLS certificate with a correctly signed one.

To log in to the web console, enter your username and password at the login screen. You can log in with the username and password of any local account on the system, including the `root` user.

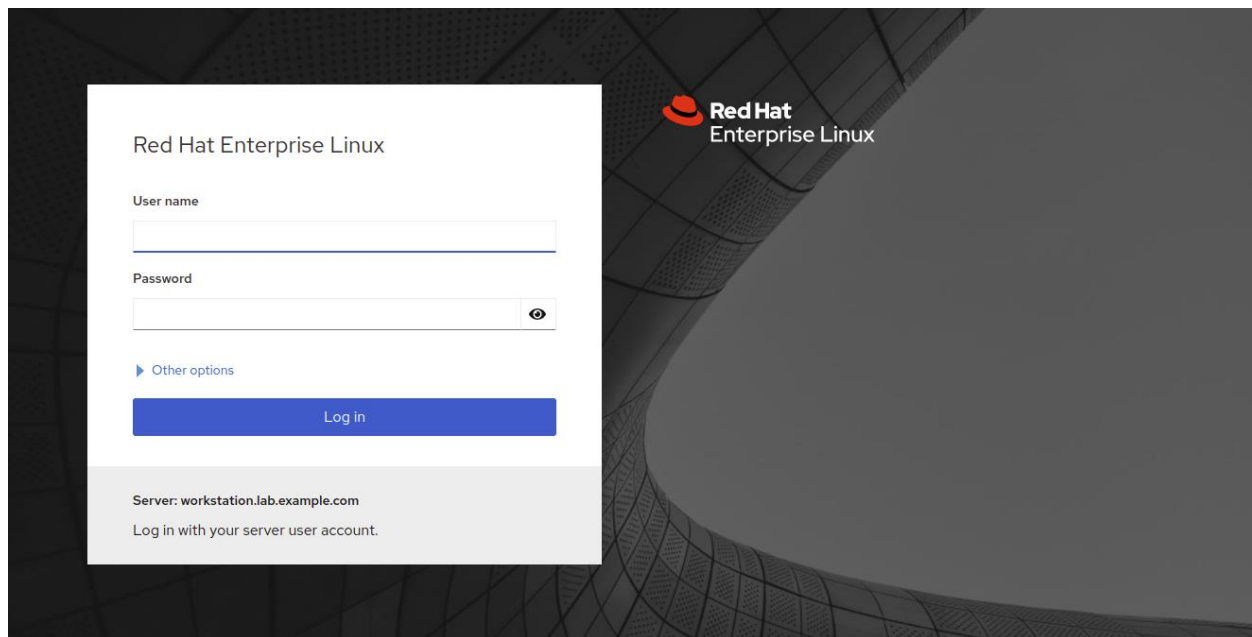


Figure 14.1: The web console login screen

Click **Log In**.

After you log in, the web console displays the username on the left side of the title bar. The default access to the web console is with limited rights, as you can see in the following **Limited access** button and in the "Web console is running in limited access mode" message.





Figure 14.2: Non-privileged user's title bar

If your account is configured with the appropriate privileges, then you can escalate privileges by switching to administrative access, by clicking the **Limited access** or **Turn on administrative access** buttons. During the escalation privileges process, you need to enter your password. When you have escalated privileges, the **Limited access** button changes to **Administrative access**.

You can switch back to limited access mode by clicking the **Administrative access** button and then clicking the **Limit access** button in the pop-up window that it shows.



Figure 14.3: Privileged user's title bar

## Change Passwords in the Web Console

You can change your own password when logged in to the web console. Click the **Accounts** button on the navigation bar. Click your account label to open the account details page.

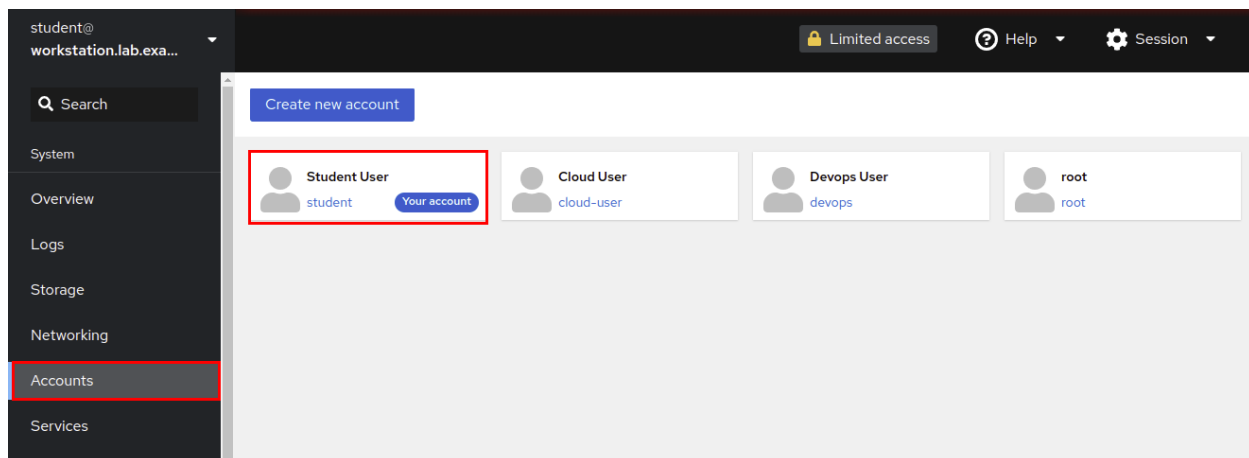


Figure 14.4: User accounts

As a non-privileged user, you are restricted to setting or resetting your password and managing public SSH keys. To set or reset your password, click the **Set password** button.

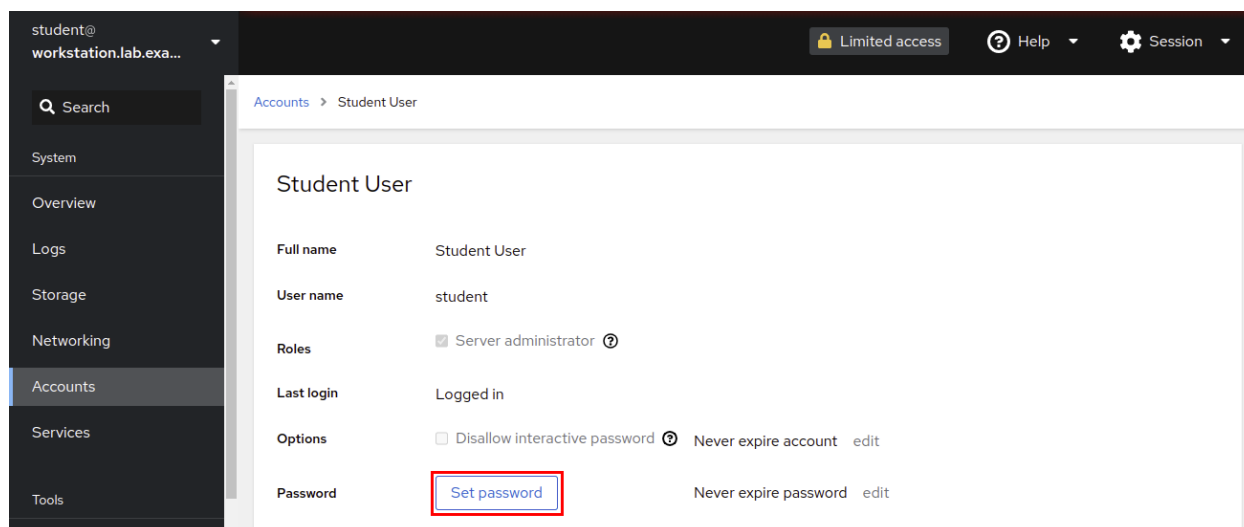


Figure 14.5: User account details

Enter your information in the **Old password**, **New password**, and **Confirm new password** fields. Click the **Set password** button to activate the new password.

## Set password

Old password

New password

Excellent password

Confirm new password

[Set password](#) [Cancel](#)

Figure 14.6: Setting and resetting passwords

## Troubleshoot with the Web Console

The web console is a powerful troubleshooting tool. You can monitor system statistics in real time, inspect system logs, and switch to a terminal session within the web console to gather additional information from the command-line interface.

### Monitor System Statistics in Real Time

Click the **Overview** button on the navigation bar to view information about the system, such as its type of hardware, operating system, hostname, and more. If you

log in as a non-privileged user, then you can see all the information but not modify any value. The following image displays the **Overview** page.

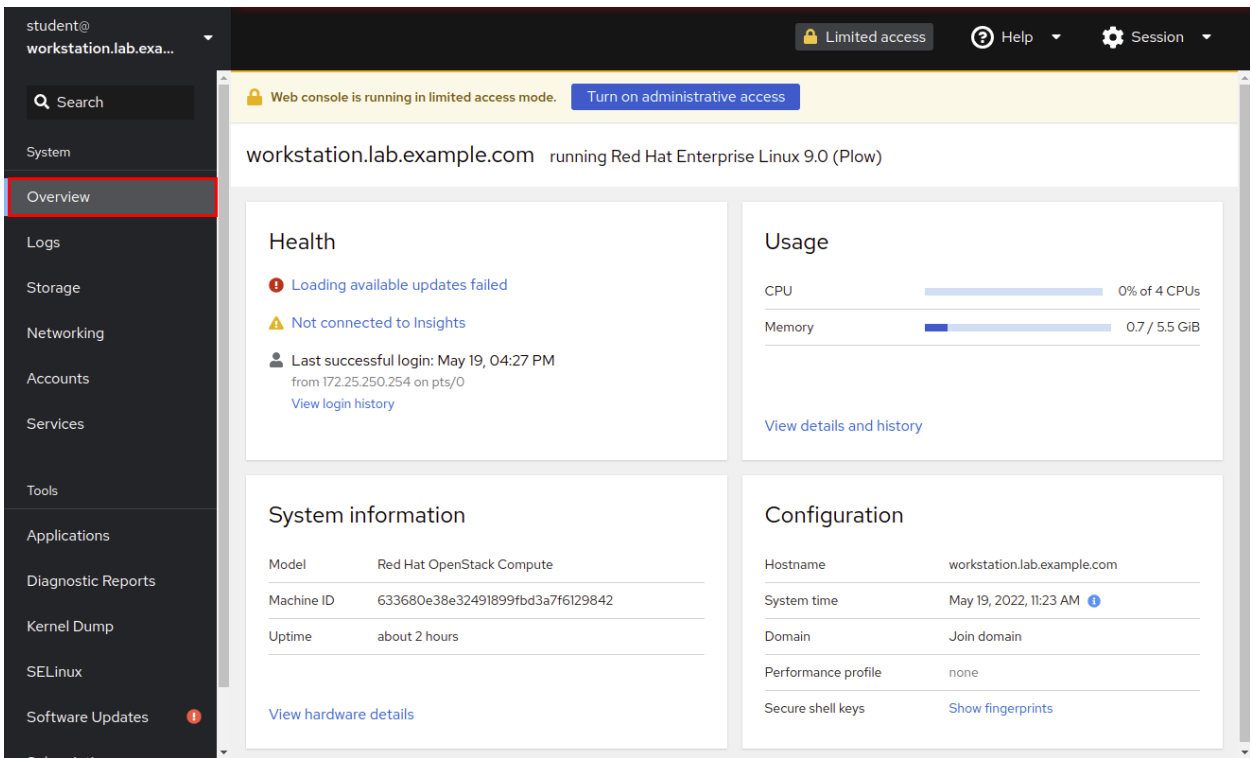


Figure 14.7: Non-privileged user's Overview page

Click **View details and history** on the **Overview** page to view details of current system performance for CPU activity, memory use, disk I/O, and network usage.

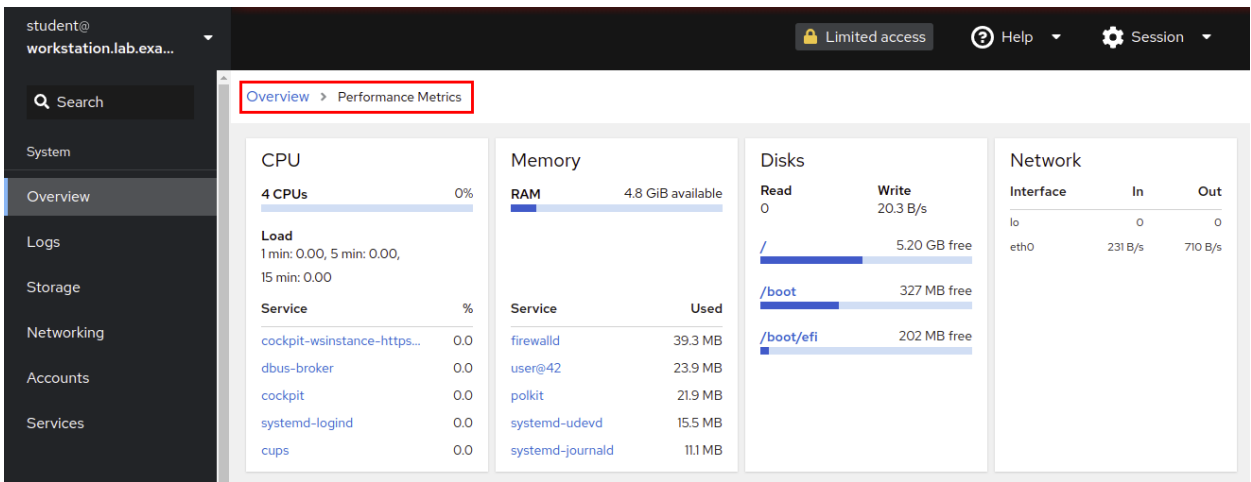


Figure 14.8: Non-privileged user's system performance metrics

Inspect and Filter Syslog Events

The **Logs** section in the navigation bar provides access to analysis tools for the system logs. You can use the scroll menus on the page to filter log messages by a logging date range, or priority, or both. The web console uses the current date as the default; you can click the date menu and specify any range of dates. Similarly, the **Priority** menu provides options that range from **Debug and above** (at the lowest level) to more specific severity conditions such as **Alert and above** or **Error and above**.

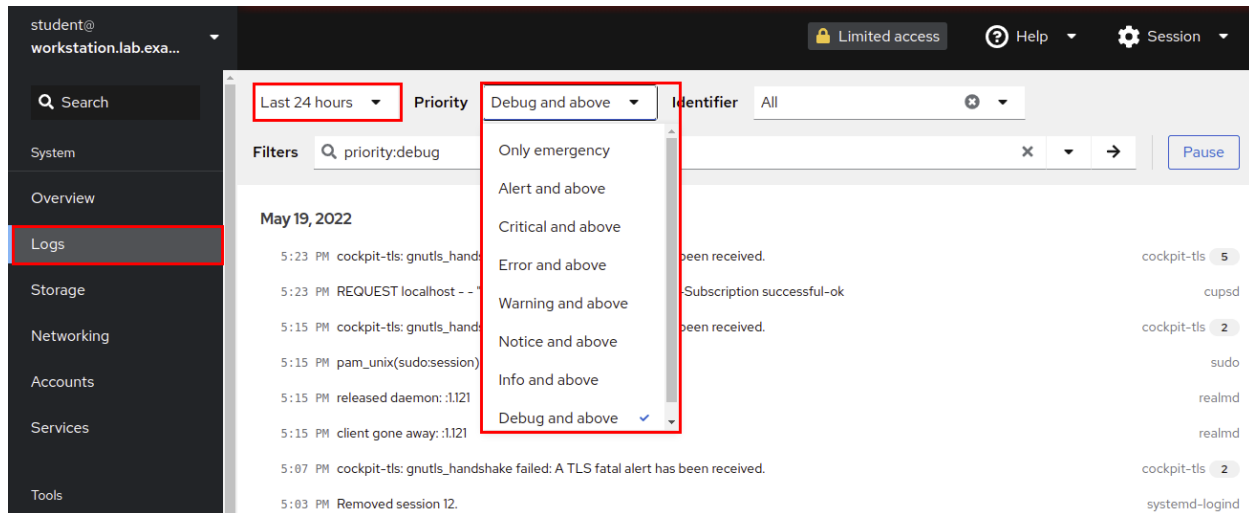


Figure 14.9: Log severity selections

Click a row to view details of the log report. In the following example, note the first row that reports on a `sudo` log message.

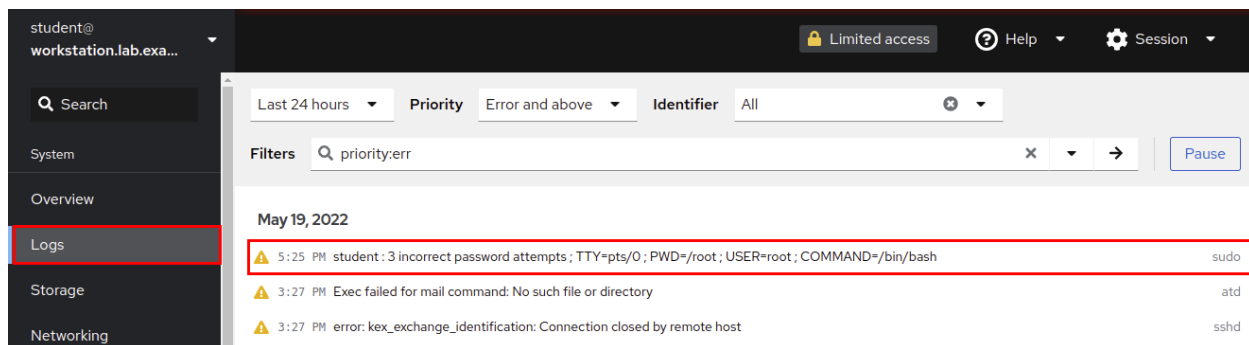


Figure 14.10: Log entry selection

The following example shows the details that the web console displays when you click the `sudo` row. Details of the report include the selected log entry (`sudo`), the date, time, priority, and syslog facility of the log entry, and the hostname of the system that reported the log message.

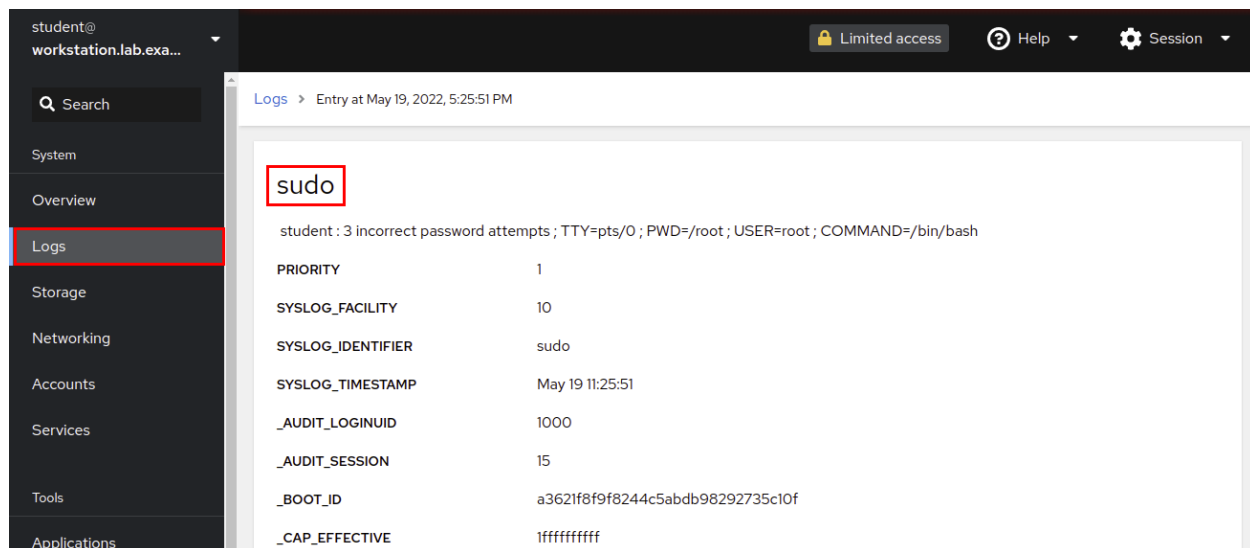


Figure 14.11: Log entry details

## Run Commands from a Terminal Session

The **Terminal** button in the navigation bar provides access to a fully functional terminal session within the web console interface. In this web console terminal, you can run arbitrary commands to manage and work with the system, and for tasks that the other web console tools do not support.

The following image displays examples of commands to gather additional information. For example, listing the contents of the `/var/log` directory provides reminders of log files that might have valuable information. The `id` command provides information such as group membership that might help to troubleshoot file access restrictions. The `ps -au` command provides a view of processes that are running in the terminal and of the user that is associated with the process.

The screenshot shows a web console interface for a Red Hat system. On the left is a navigation sidebar with categories like Networking, Accounts, Services, Tools, Applications, Diagnostic Reports, Kernel Dump, SELinux, Software Updates, and Subscriptions. The 'Terminal' option at the bottom is highlighted with a red box. The main area displays a terminal session for a user named 'student' at 'workstation.lab.ex...'. The terminal shows the command `ls /var/log` being executed, resulting in a list of log files such as `audit`, `boot.log`, `chrony`, `cloud-init.log`, `cloud-init-output.log`, `cron`, `cups`, `dnf.librepo.log`, `dnf.log`, `dnf.rpm.log`, `firewalld`, `gdm`, `hawkey.log`, `heat-provision.log`, `insights-client`, `kdump.log`, `lastlog`, `maillog`, `messages`, `part-handler.log`, `private`, `qemu-ga`, `README`, `rhsm`, `samba`, `secure`, `speech-dispatcher`, `spooler`, `sssd`, `tallylog`, `tuned`, and `wtmp`. The `ls` command and its output are enclosed in a red box. Below this, the `id` command is executed, showing the user's identity: `uid=1000(student) gid=1000(student) groups=1000(student),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023`. This output is also enclosed in a red box. Finally, the `pwd` command is executed, showing the current directory as `/home/student`. The terminal prompt is `[student@workstation ~]$`.

Figure 14.12: Non-privileged terminal session troubleshooting  
Create Diagnostic Reports

A diagnostic report is a collection of configuration details, system information, and diagnostic information from a Red Hat Enterprise Linux system. Data that is collected in the report includes system logs and debug information that you can use to troubleshoot issues.

To generate a diagnostic report, log in to the web console as a privileged user. Click the **Diagnostic Reports** button on the navigation bar to open the page that creates these reports. Click the **Create report** button to generate a new diagnostic report.

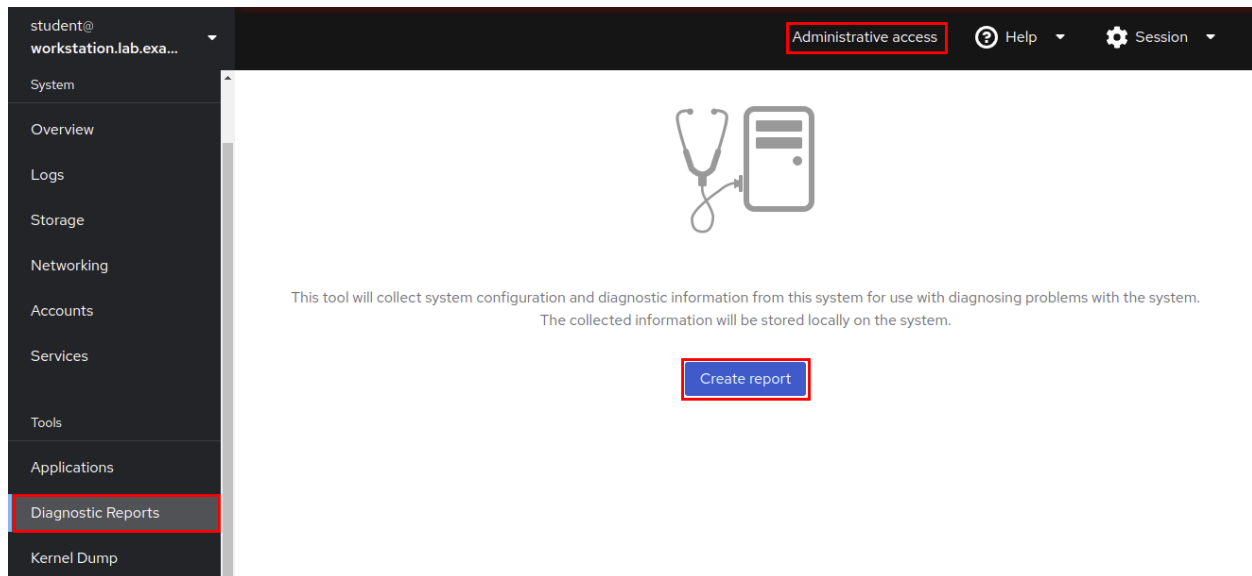


Figure 14.13: Create a diagnostic report

After some minutes, the interface displays **Done!** when the report is complete. Click the **Download report** button to save the report to your local system.



Figure 14.14: Download a completed report

Click **Save File** and complete the process.

## Manage System Services with the Web Console

As a privileged web console user, you can stop, start, enable, and restart system services. Additionally, you can configure network interfaces, configure firewall services, administer user accounts, and more.

## System Power Options

In the web console, you can restart or shut down the system. To access the system power options, log in to the web console as a privileged user. Click the **Overview** button on the navigation bar to access system power options.

From the menu on the upper right, select the appropriate option to either reboot or shut down a system.

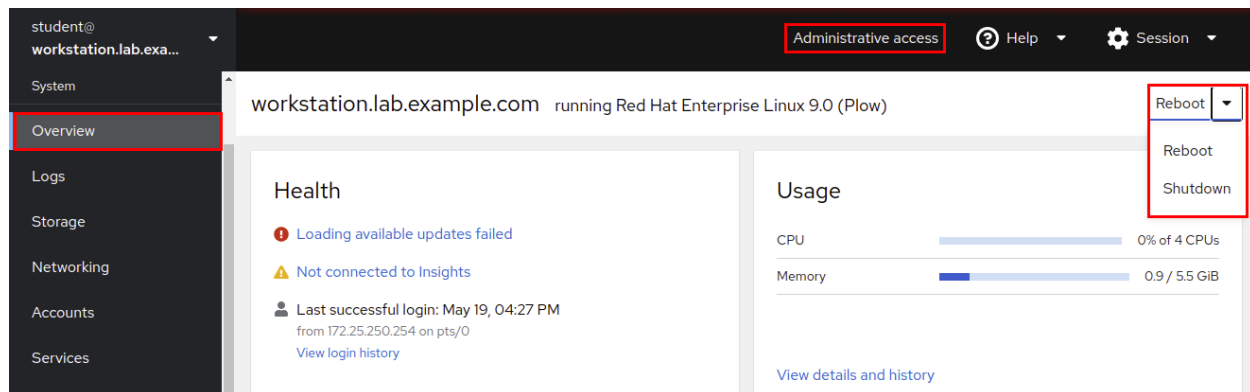


Figure 14.15: System power options

## Control Running System Services

You can start, enable, disable, and stop services with graphical tools in the web console. To do so, click the **Services** button on the navigation bar to access the web console's services initial page. The **Services** page shows the system services tab by default. You can change to **Targets** or **Sockets** by clicking the appropriate tab. Use the search bar or scroll through the page to select the service to manage.

In the following example, select the `atd.service` row to open the service management page.



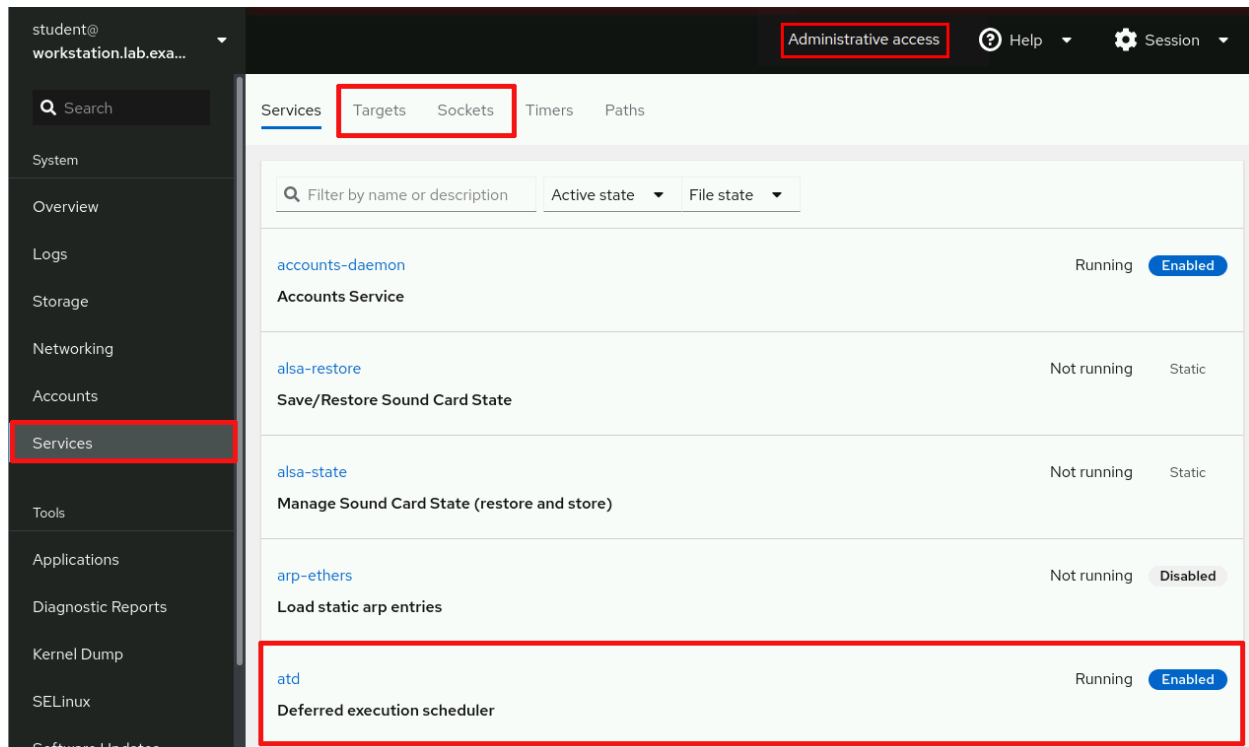


Figure 14.16: Services: Initial view

Click the **Stop**, **Restart**, or **Disallow running (mask)** buttons as appropriate to manage the service. In this view, the service is already running. To view additional information about the service, click any of the highlighted links or scroll through the service logs that are displayed below the service management section.

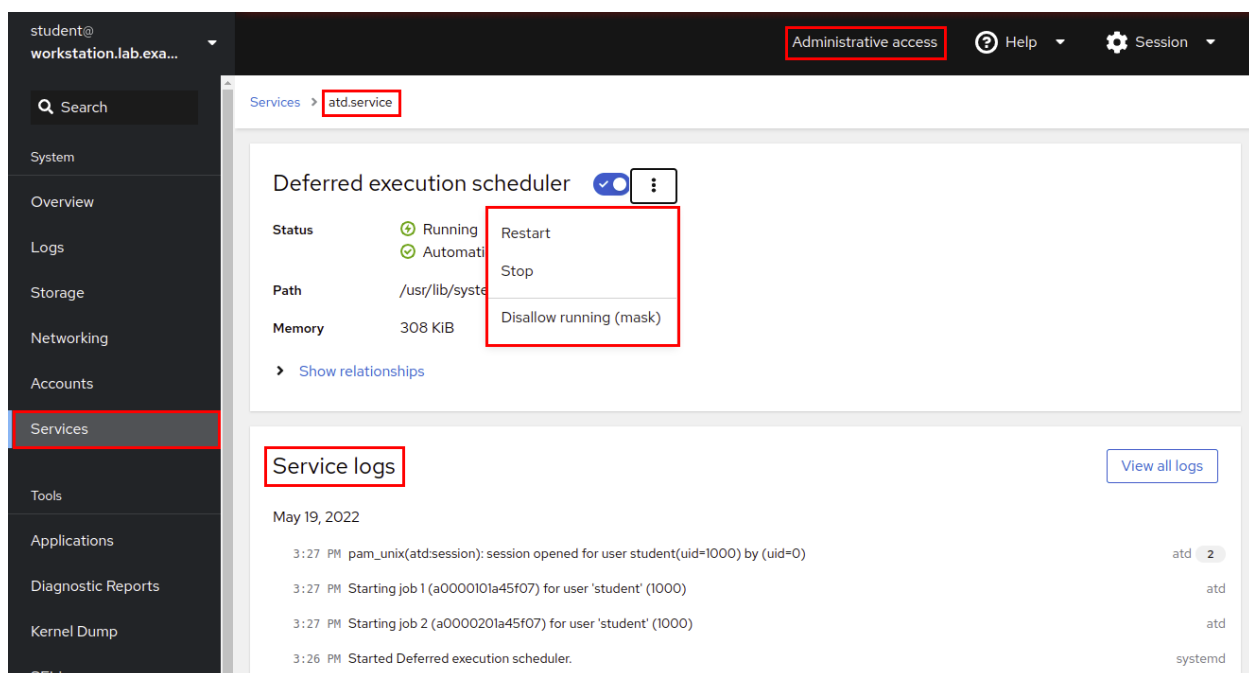


Figure 14.17: Services: Service details and management interface

## Configure Network Interfaces and the Firewall

To manage firewall rules and network interfaces, click the **Networking** button on the navigation bar. The following example shows how to gather information about network interfaces and how to manage them.

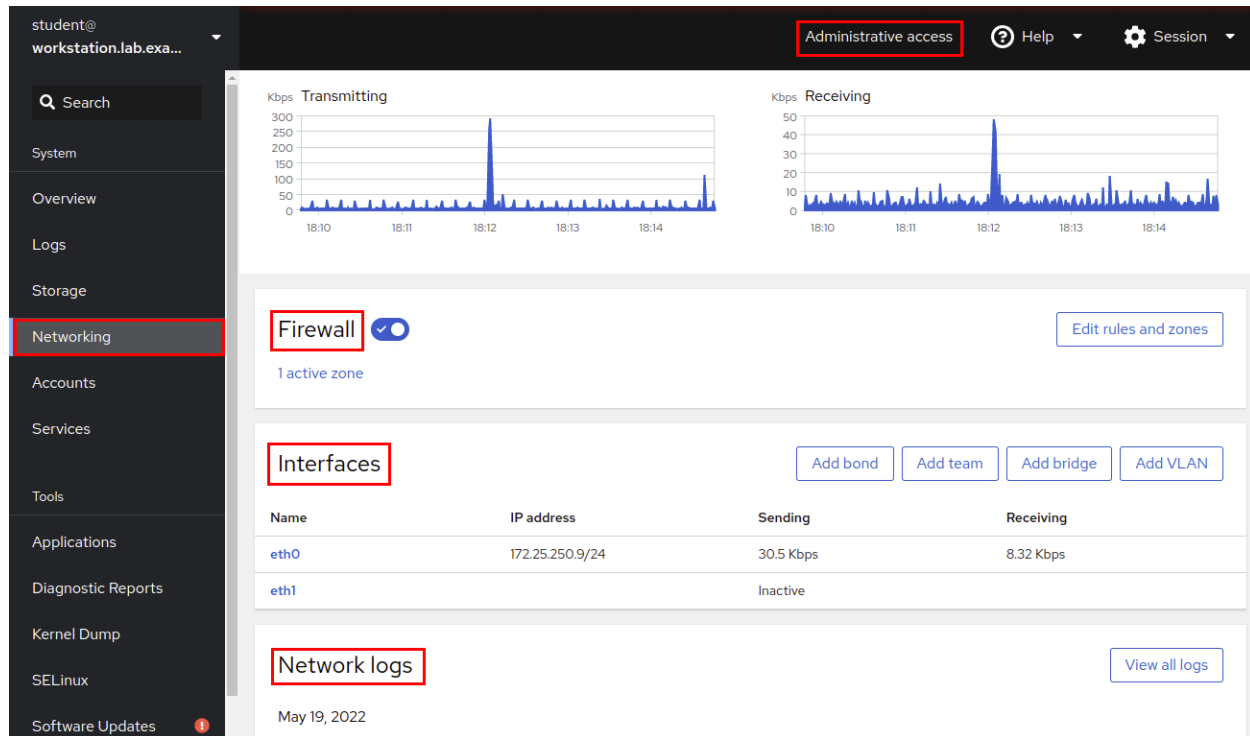


Figure 14.18: Networking: Initial view

Click the appropriate interface name in the **Interfaces** section to access the management page. In this example, the `eth0` interface is selected. The top part of the management page displays network traffic activity for the selected device. Scroll down to view configuration settings and management options.

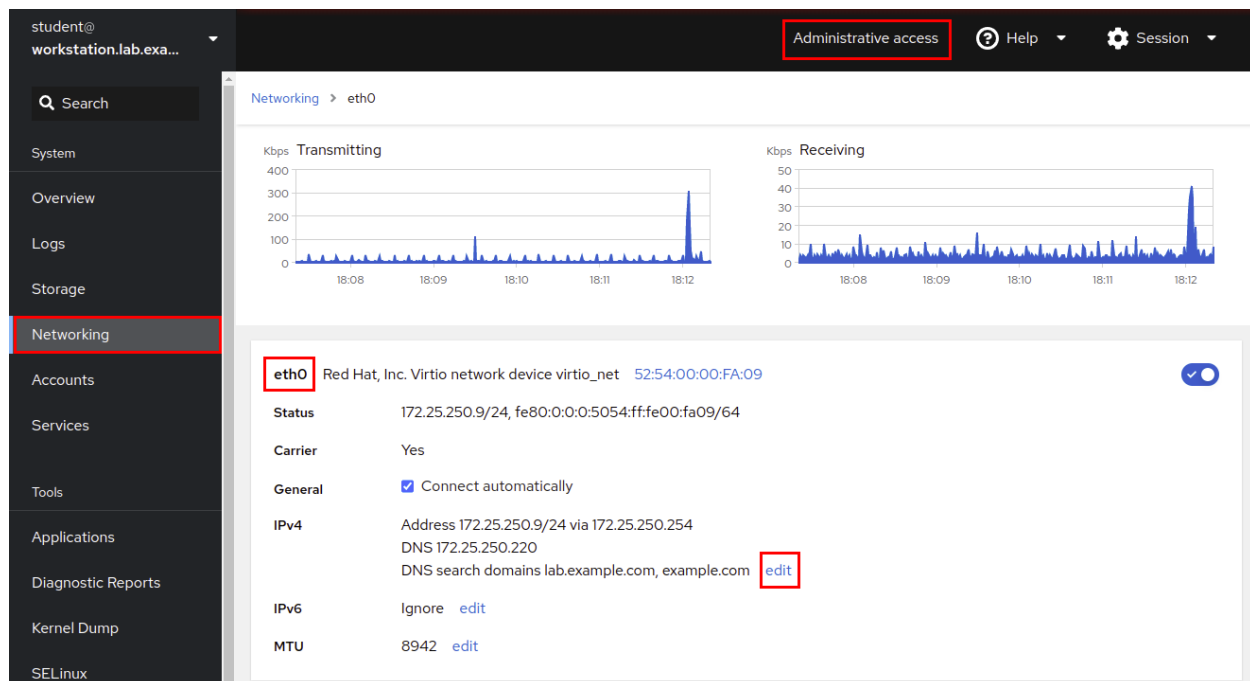


Figure 14.19: Networking: Interface details

To modify or add configuration options in an interface, click the highlighted links for the wanted configuration. In this example, the **IPv4** link shows a single IP address and netmask, 172.25.250.9/24 for the `eth0` network interface. To add an IP address to the `eth0` network interface, click the **edit** link.

Click the **+** symbol on the right side of the **Manual** list selection to add an IP address. Enter an IP address and network mask in the appropriate fields. Click **Apply** to activate the new settings.

## IPv4 settings

Addresses

Manual



Address

Prefix length or netmask

Gateway

172.25.250.9

24

172.25.250.254



Address

Prefix length or netmask

Gateway

172.25.250.100

24



DNS



Automatic



Server

172.25.250.220



DNS search domains



Automatic



Search domain

Apply

Cancel

Figure 14.20: Add an IP address to an existing interface

The display automatically switches back to the interface's management page where you can confirm the new IP address.

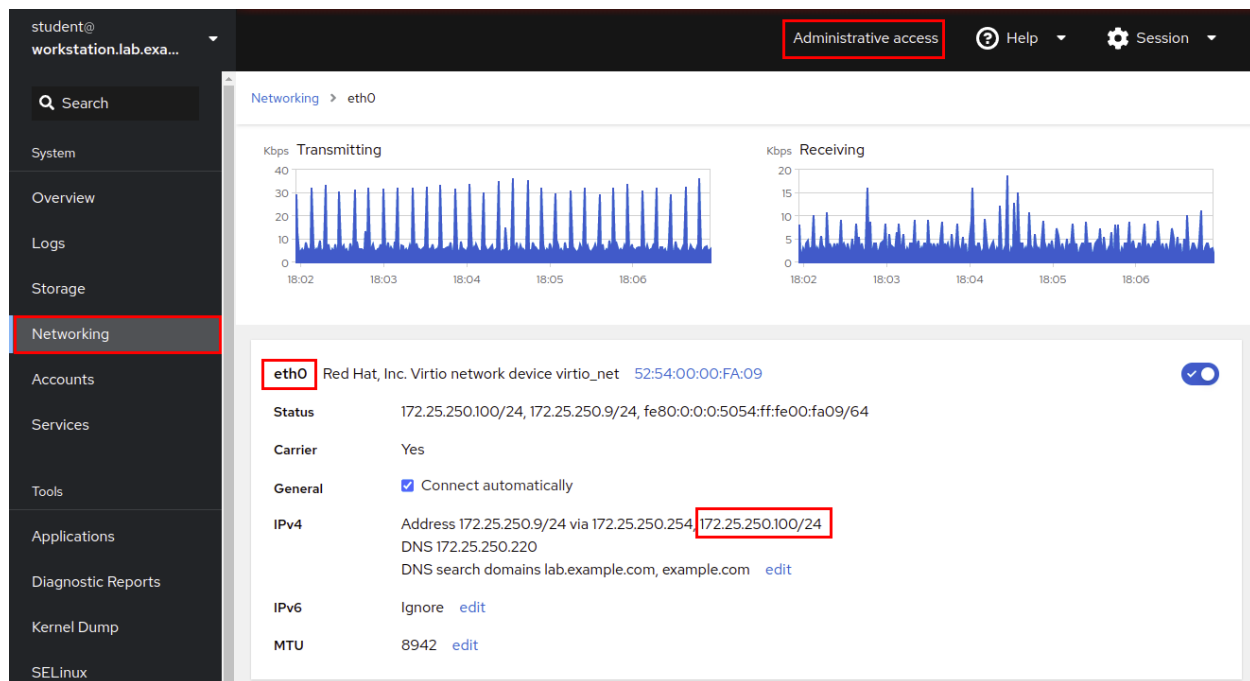


Figure 14.21: Confirm the new IP address

## Administer User Accounts

As a privileged user, you can create user accounts in the web console.

Click **Accounts** on the navigation bar to view existing accounts. Click **Create new account** to open the account management page.

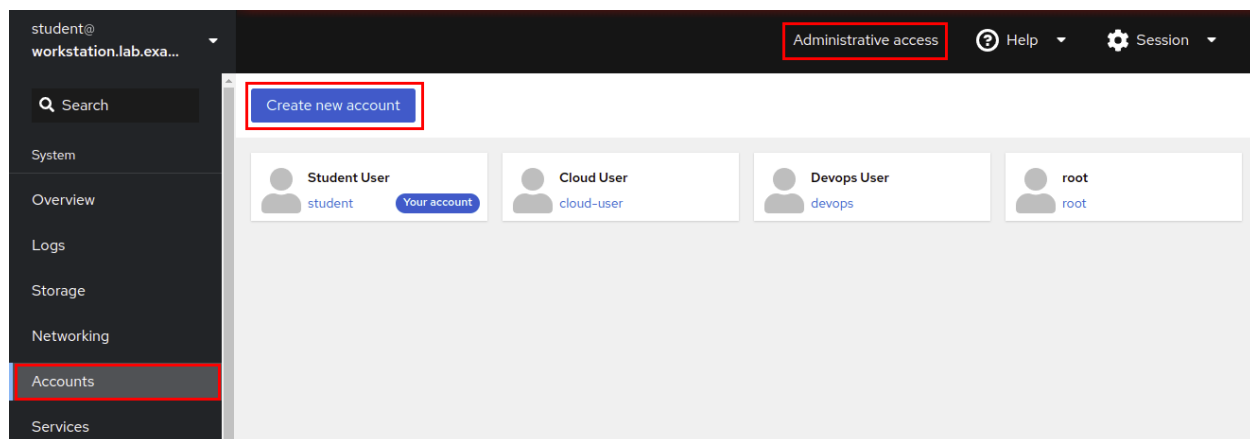


Figure 14.22: Existing user accounts

Enter the information for the new account and then click **Create**.

## Create new account

Full name	<input type="text" value="New User"/>
User name	<input type="text" value="nuser"/>
Password	<input type="password" value="....."/> <div>Excellent password</div>
Confirm	<input type="password" value="....."/>
Options	<input type="checkbox"/> Disallow interactive password ?
<div><div>Create</div><div>Cancel</div></div>	

Figure 14.23: Create an account

The display automatically reverts to the account management page, where you can confirm the new user account.

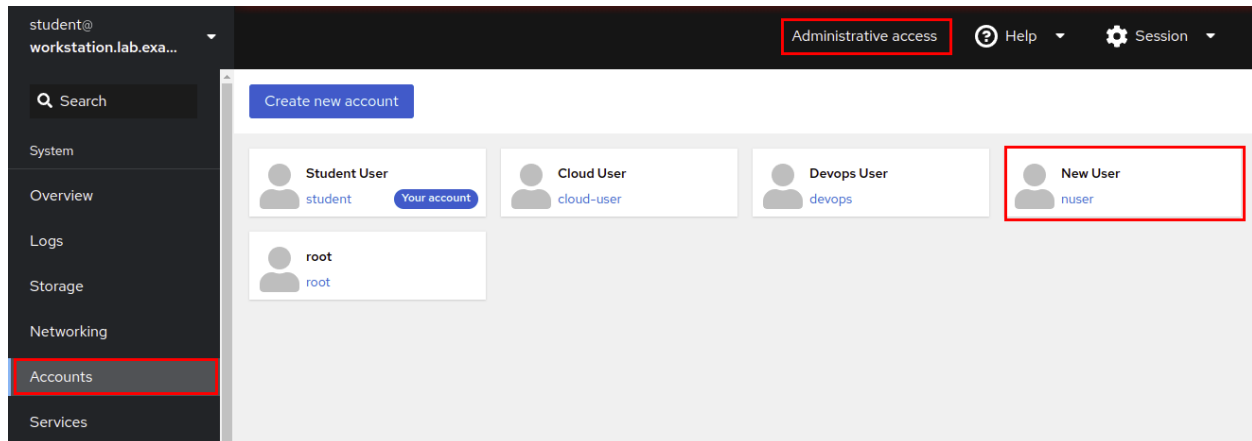


Figure 14.24: Account management page

## References

cockpit(1), cockpit-ws(8), and cockpit.conf(5) man pages

For more information, refer to *Managing Systems Using the RHEL 9 Web Console* at <https://access.redhat.com/documentation/en->

[us/red hat enterprise linux/9/html-single/managing\\_systems\\_using\\_the\\_rhel\\_9\\_web\\_console/index#getting-started-with-the-rhel-9-web-console](https://us.redhatenterprise.com/linux/9/html-single/managing_systems_using_the_rhel_9_web_console/index#getting-started-with-the-rhel-9-web-console) managing-systems-using-the-web-console

[Next](#)

## Guided Exercise: Analyze and Manage Remote Servers

In this exercise, you enable and access the web console on a server to manage it and to diagnose and resolve issues.

### Outcomes

- Use the web console to monitor system features, inspect log files, create user accounts, and access the terminal.

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 support-cockpit
```

### Instructions

1. Log in to the servera machine as the student user.

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

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

3. The web console is already installed on the system, but it is not active. Enable and start the cockpit service.

1. Enable the web console service.

```
2. [student@servera ~]$ sudo systemctl enable --now cockpit.socket
```

```
3. [sudo] password for student: student
```

```
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket -> /usr/lib/systemd/system/cockpit.socket.
```

4. On the workstation machine, open the Firefox web browser and log in to the web console interface at `servera.lab.example.com`. Log in as the student user.
  1. Open the browser and navigate to `https://servera.lab.example.com:9090`.
  2. Accept the self-signed certificate by adding it as an exception.
  3. Log in as the student user, with `student` as the password.

You are now logged in to the web console as a normal user, with minimal privileges.

5. Verify your current authorization within the web console interface.
  1. Click the **Terminal** button on the left navigation bar to access the terminal.

A terminal session opens where the student user is already logged in. Verify that command execution works in the embedded terminal.

```
[student@servera ~]$ id
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

2. Click the **Accounts** button on the left navigation bar to manage users.

The **Create new account** button does not appear, because you are logged in with limited access.

3. Click the **Student User** link.

On the student user's account details page, you can only set a new password or add an authorized SSH public key.

6. Escalate privileges in the web console.
  1. Click the **Limited access** button to switch to administrative access. Use `student` as the student user password and click the **Authenticate** button. The web console replaces the **Limited access** button by the **Administrative access** button.
7. To investigate system statistics, click **Overview** on the left navigation bar and the **View details and history** button on the **Usage** section.



This page displays various operating system statistics, such as current load, disk usage, disk I/O, and network traffic.

8. To inspect system logs, click the **Logs** button on the left navigation bar.

This page displays the `systemd` system logs. Use the buttons in the upper-left corner to modify how the web console displays the log entries based on date and the priority of the logs.

1. Click the **Priority** list and choose **Debug and above**.
  2. Based on the current day of the month, click any log entry from the list. A log entry detail page opens with additional information about the event, such as the hostname, the SELinux context, or the PID number of the process that the entry corresponds to.
9. Add a second IP address to an existing network interface device.
  1. Click the **Networking** button on the left navigation bar.

This page displays details of the current network configuration for `servera`, as well as real-time network statistics, firewall configuration, and log entries about networking.

2. Scroll down to the **Interfaces** section and click the row for the `eth0` network interface.

A details page displays real-time network statistics, as well as the current configuration for that network interface.

3. Click the **edit** link in the **IPv4** section.

An **IPv4 settings** window opens, where you can change the network interface configuration.

4. In the **IPv4 settings** window, click the **+** button next to the **Manual** list.
  5. In the **Address** text box, enter `172.25.250.99` as the second IP address.
  6. In the **Prefix length or Netmask** text box, enter `24` as the netmask value.
  7. Click **Apply** to save the new network configuration.

The new configuration is applied immediately in the web console. The new IP address is visible in the **IPv4** line.

10. Create a user account.

1. Click the **Accounts** button on the left navigation bar. The web console now shows the **Create new account** button, because you have administrative rights.
2. Click the **Create new account** button.
3. In the **Create new account** window, add the following details:

<i>Field</i>	<i>Value</i>
Full Name	manager1
User Name	manager1
Password	redh@t!23
Confirm	redh@t!23

4. Click **Create**.

11. Access a terminal session within the web console to add the manager1 user to the wheel group.

1. Click the **Terminal** button on the left navigation bar.
2. Use the `id manager1` command to view the group membership of the manager1 user.

```
3. [student@servera ~]$ id manager1
```

```
uid=1002(manager1) gid=1002(manager1) groups=1002(manager1)
```

4. Use the `sudo usermod -aG wheel manager1` command to add the manager1 user to the wheel group.

```
5. [student@servera ~]$ sudo usermod -aG wheel manager1
```

```
[sudo] password for student: student
```

6. Use the `id manager1` command again to verify that the manager1 user is a member of the wheel group.

```
7. [student@servera ~]$ id manager1
```

```
uid=1002(manager1) gid=1002(manager1) groups=1002(manager1),10(wheel)
```

12. Enable and start the Kernel process accounting service (psacct).

1. Click the **Services** button on the left navigation bar.

2. Search for the **Kernel process accounting** service. Click the service link.  
A details page displays the service status as disabled.
  3. Click the **Start and Enable** button next to the service name.
  4. The service is now enabled and started.
13. Log off from the web console interface.
  14. Return to the workstation system as the student user.

```
15. [student@servera ~]$ exit
```

```
[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 support-cockpit
```

This concludes the section.

[Previous](#) [Next](#)

# Create a Diagnostics Report

## Objectives

Describe and use Red Hat Customer Portal key resources to find information from Red Hat documentation and the Knowledgebase.

## Resources on the Red Hat Customer Portal

The Red Hat Customer Portal at <https://access.redhat.com> gives customers access to documentation, downloads, tools, and technical expertise. The Knowledgebase enables customers to search for solutions, FAQs, and articles. The following list shows some functions of the Red Hat Customer Portal:

- Access official product documentation, solutions, and FAQs.

- Submit and manage support cases.
- Manage software subscriptions and entitlements.
- Obtain software downloads, updates, and evaluations.
- Access a catalog of security advisories for Red Hat products.
- Access an integrated search engine for Red Hat resources.
- Access white papers, information sheets, and multimedia presentations.
- Participate in community discussions.

Parts of the site are publicly accessible, whereas other areas require an active subscription. Visit <https://access.redhat.com/help/> for help with accessing the Red Hat Customer Portal.

## Tour of the Red Hat Customer Portal

Access the Red Hat Customer Portal by visiting <https://access.redhat.com/>. This section introduces the Red Hat Customer Portal tour at <https://access.redhat.com/start>.

With the tour, you can discover portal features and maximize the benefits of your Red Hat subscription. After you log in to the Red Hat Customer Portal, click the **Tour the Customer Portal** button.

The **WELCOME TO THE RED HAT CUSTOMER PORTAL** window appears. Click the **Let's go** button to start the tour.

### The Top Navigation Bar

The first menus on the tour, on the top navigation bar, are Subscriptions, Downloads, Containers, and Support Cases.

The **Subscriptions** menu opens a new page to manage your registered systems, subscriptions, and entitlements. This page lists applicable errata information. You can create activation keys to register systems and to ensure correct entitlements. The Organization Administrator for your account might restrict your access to this page.

The **Downloads** menu opens a new page to access your product downloads and to request evaluation for the products with no entitlements.

The **Support Cases** menu opens a new page to create, track, and manage your support cases through the Case Management system, if authorized by your organization.

With the **User Menu** menu, manage your account, any accounts for which you are an Organization Administrator, your profile, and email notification options.

The globe icon opens the **Language** menu to specify your language preferences for the Red Hat Customer Portal.

### Navigate the Red Hat Customer Portal Menus

Underneath the top navigation bar on the main page are menus to navigate to major categories of resources on the site.

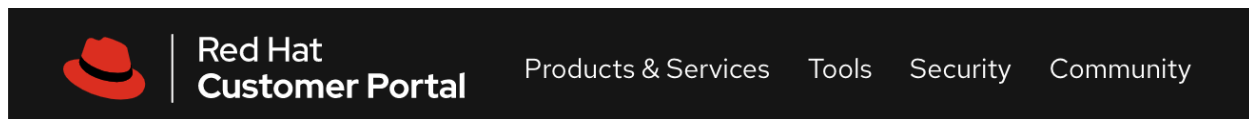


Figure 14.25: Red Hat Customer Portal Menus

The **Products & Services** menu gives access to the Product Hubs, for product-specific evaluations, getting started guides, and other product support information. You can also access documentation for Red Hat products, a knowledge base of support articles, and support policies, and can contact Red Hat Support. You can access services that Red Hat provides, such as Consulting, Technical Account Management, and Training and Certifications.

The **Tools** menu provides tools to help you to succeed with Red Hat products. The tools help to troubleshoot a product issue, and provide packages and errata information. The **Customer Portal Labs** section has a collection of web-based applications and tools to help you to improve performance, diagnose issues, identify security problems, and optimize your configurations. The **Red Hat Insights** section helps to analyze platforms and applications to predict risk, take recommended actions, and track costs to manage hybrid cloud environments. Insights alert administrators before an outage, or about a security event or overspending.

The **Security** menu provides access to the Red Hat Product Security Center for security updates and prevents environments from exposure to security vulnerabilities. This section provides information about high-profile security issues, with access to the security advisories, the Red Hat *Common Vulnerabilities and*

*Exposures (CVE) database, the security labs, the Red Hat security blog, security measurement, severity ratings, backporting policies, and product signing GNU Privacy Guard (GPG) keys.*

The **Community** menu gives access to the Customer Portal Community section for discussions and private groups. This section is a place where Red Hat experts, customers, and partners communicate and collaborate. This section contains discussion forums, blogs, and information about upcoming events.

## Note

Red Hat recommends viewing the complete tour at [Getting Started with Red Hat](#), including the sections on the **How to Personalize Your Customer Portal experience** menu, the **Explore the Benefits of Your Red Hat subscription** menu, and the **How to Engage Red Hat Support** menu. An active subscription is required to access these subscription resources.

## Contact Red Hat Customer Support

The Red Hat Customer Portal provides access to technical support for customers with an active subscription. You can contact support by opening a support case or a chat session, or by phone. For detailed information, visit the [https://access.redhat.com/support/policy/support\\_process](https://access.redhat.com/support/policy/support_process) address.

## Prepare a Support Case

Before contacting Red Hat support, it is important to gather relevant information for the report.

*Define the problem.* State the problem and its symptoms specifically. Provide detailed steps to reproduce the problem.

*Gather background information.* Which product and version are affected? Be ready to provide relevant diagnostic information. This information might include the output of the `sos report` command. For kernel problems, the information might consist of the system's `kdump` crash dump or a digital photo of the kernel backtrace that is displayed on the monitor of a crashed system.

*Determine the severity level.* Red Hat uses four severity levels to classify issues. *Urgent* and *High* severity problem reports must be followed by a phone call

to the relevant local support center  
(see <https://access.redhat.com/support/contact/technicalSupport>).

Severity	Description
Urgent (Severity 1)	A problem that severely impacts your use of the software in a production environment. This severity includes loss of production data or malfunctioning production systems. The situation halts your business operations, and no procedural workaround exists.
High (Severity 2)	A problem where the software is functioning but its use in a production environment is severely reduced. The situation is causing a high impact on your business operations, and no procedural workaround exists.
Medium (Severity 3)	A problem that involves partial, non-critical loss of use of the software in a production environment or development environment. The problem involves a medium to low impact on your business for production environments. The business continues to function by using a procedural workaround. For development environments, the situation is causing problems with migrating your project into production.
Low (Severity 4)	A general usage question, reporting of a documentation error, or recommendation for a future product enhancement or modification. The problem involves low to no impact on your business or the performance or functioning of your system. The problem involves medium to low impact on your business for development environments, but your business continues to function by using a procedural workaround.

### The sos Report Utility

The sos report is generally the starting point for Red Hat technical support to investigate the reported problem. This utility provides a standardized way to collect diagnostic information that Red Hat technical support needs for investigating the reported issues. The `sos report` command collects various debugging information from one or more systems, and provides an option to remove sensitive data. This report is attached to the Red Hat support case. The `sos collect` command runs and collects individual sos reports from a specified set of nodes. The `sos clean` command obfuscates potentially sensitive information such as usernames, hostnames, IP or MAC addresses, or other user-specified data.

The following list contains information that can be collected in a report:

- The running kernel version
- Loaded kernel modules
- System and service configuration files

- Diagnostic command output
- A list of installed packages

You can generate a diagnostics report to submit to Red Hat technical support by using either the web console or the command line.

### Generate an sos Report with the Web Console

To generate an `sos` report with the web console, you must log in as a privileged user.

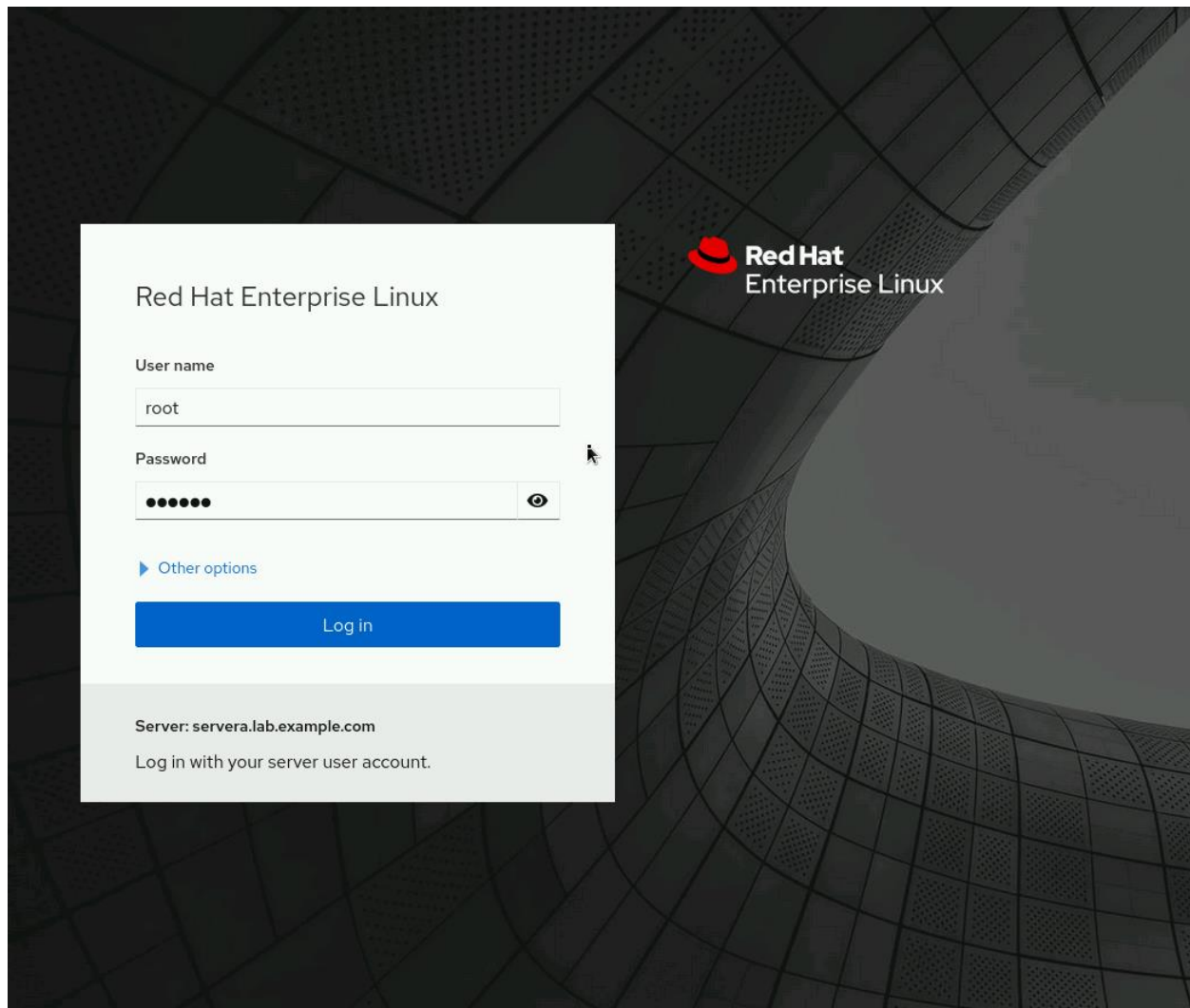


Figure 14.26: Use a privileged user to log in  
Click **Diagnostic Reports** and then click **Create report**:



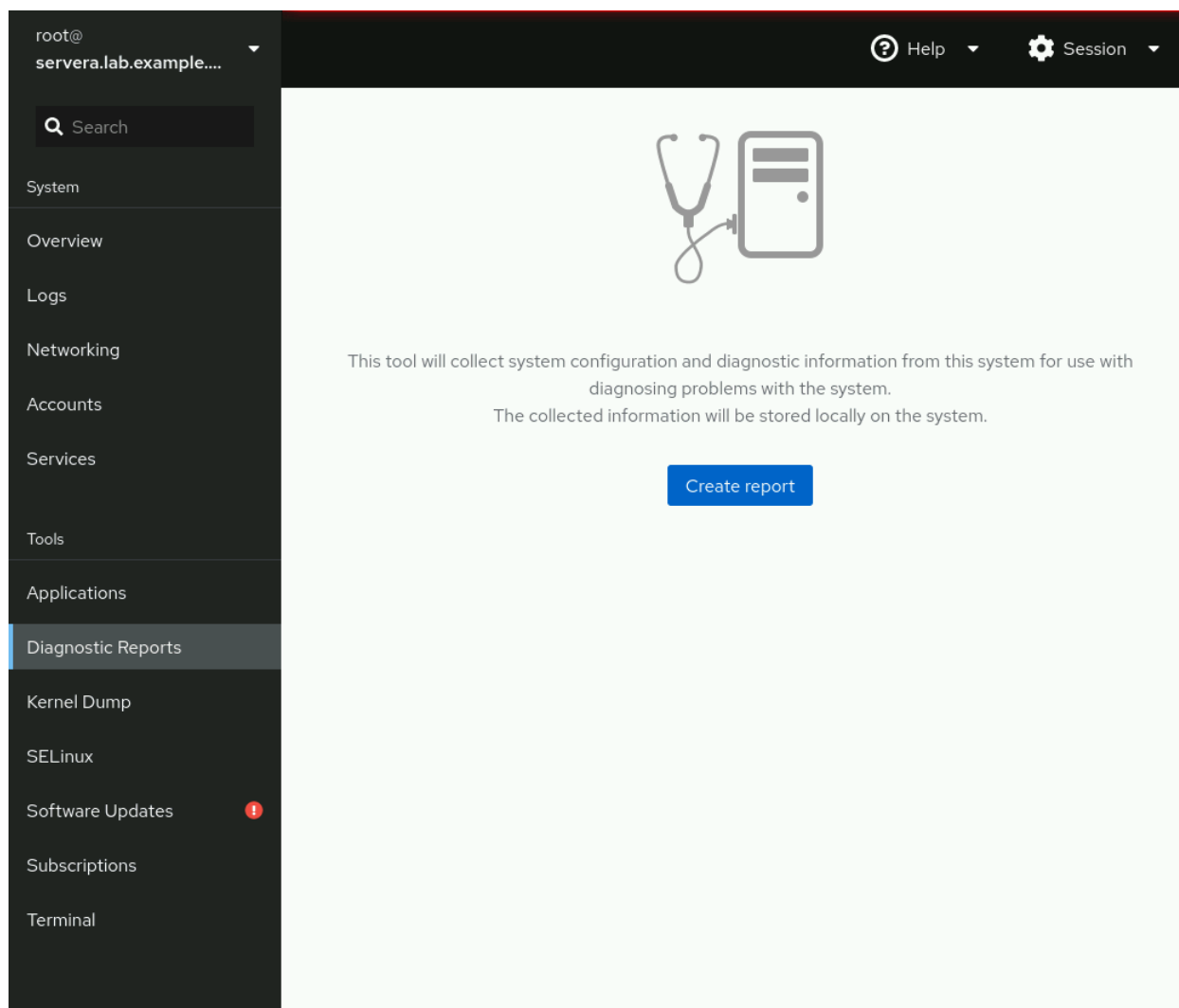


Figure 14.27: Create a diagnostic report

The diagnostic report takes a few minutes to generate.

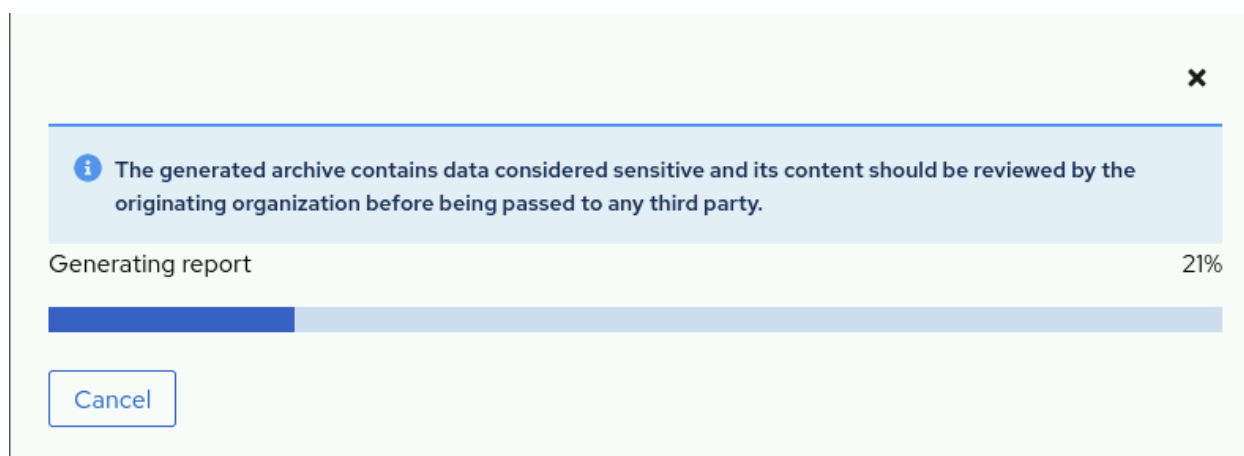


Figure 14.28: Generate a diagnostic report

Click **Download report** followed by **Save File** to save the diagnostic report.



Figure 14.29: Download the diagnostic report to your local system  
Generate an sos Report from the Command Line

Red Hat Enterprise Linux installs the `sos` report utility with the `sos` package:

```
[root@host ~]# dnf install sos
...output omitted...
Complete!
```

Generating the `sos` report requires root privileges. Run the `sos report` command to generate the report.

```
[root@host ~]# sos report
...output omitted...
Press ENTER to continue, or CTRL-C to quit.

Optionally, please enter the case id that you are generating this report for []:
...output omitted...

Your sosreport has been generated and saved in:

  /var/tmp/sosreport-host-2022-03-29-wixbhpz.tar.xz
..output omitted...

Please send this file to your support representative.
```

When you provide any support case ID in the previous command, the report attaches directly to the previously created support case. You can also use the `sos report` command `--utility` option to send the report to technical support.

Verify that the `sos report` command created the archive file at the previous location.

```
[root@host ~]# ls -l /var/tmp/
total 9388
-rw-----. 1 root root 9605952 Mar 29 02:09 sosreport-host-2022-03-29-wixbhpz.tar.xz
-rw-r--r--. 1 root root      65 Mar 29 02:09 sosreport-host-2022-03-29-wixbhpz.tar.xz
.sha256
...output omitted...
```

The `sos clean` command obfuscates personal information from the report.

```
[root@host ~]# sos clean /var/tmp/sosreport-host-2022-03-29-wixbhpz.tar.xz*
...output omitted...
Press ENTER to continue, or CTRL-C to quit.
...output omitted...
The obfuscated archive is available at
    /var/tmp/sosreport-host0-2022-03-29-wixbhpz-obfuscated.tar.xz
...output omitted...
Please send the obfuscated archive to your support representative and keep the mapping file private
```

## Send an sos Report to Red Hat Technical Support

Select one of these methods to send an sos report to Red Hat Technical Support.

- Send an sos report by using the `sos report` command `--upload` option.
- Send an sos report to the Red Hat Customer Portal by attaching it with the support case.

## Join the Red Hat Developer Program

The Red Hat Developer Program at <https://developers.redhat.com> provides subscription entitlements to Red Hat software for development purposes, documentation, and premium books from experts on microservices, serverless

computing, Kubernetes, and Linux. Blog links to information about upcoming events and training and other helpful resources are also available.

For more information, visit <https://developers.redhat.com/>.

## References

sosreport(1) man page

[Contacting Red Hat Technical Support](#)

[Help - Red Hat Customer Portal](#)

For further information, refer to *Generating an SOS Report for Technical Support* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/9/html-single/getting\\_the\\_most\\_from\\_your\\_support\\_experience/generating-an-sos-report-for-technical-support\\_getting-the-most-from-your-support-experience](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9/html-single/getting_the_most_from_your_support_experience/generating-an-sos-report-for-technical-support_getting-the-most-from-your-support-experience)

[Previous](#) [Next](#)

# Detect and Resolve Issues with Red Hat Insights

## Objectives

Use Red Hat Insights to analyze servers for issues, remediate or resolve them, and confirm that the solution worked.

## Introduction to Red Hat Insights

Red Hat Insights is a predictive analytics tool to help you to identify and remediate threats to security, performance, availability, and stability on systems in your infrastructure that run Red Hat products. Red Hat delivers Red Hat Insights as a Software-as-a-Service (SaaS) product, so that you can deploy and scale it with no additional infrastructure requirements. In addition, you can take advantage of the

latest recommendations and updates from Red Hat that apply to your deployed systems.

Red Hat regularly updates the knowledge base, based on common support risks, security vulnerabilities, known-bad configurations, and other issues that Red Hat identifies. Red Hat validates and verifies the actions to mitigate or remediate these issues. With this support, you can proactively identify, prioritize, and resolve issues before they become a larger problem.

For each detected issue, Red Hat Insights provides risk estimates and recommendations on how to mitigate or remediate the problem. These recommendations might suggest materials such as Ansible Playbooks, or provide step-by-step instructions to help you to resolve the issue.

Red Hat Insights tailors recommendations to each system that you register to the service. To start using Red Hat Insights, install the agent in each client system to collect metadata about the runtime configuration of the system. This data is a subset of what you might provide to Red Hat Support by using the `sosreport` command to resolve a support ticket.

You can limit or obfuscate the data that your client systems send. By limiting the data, you might block some analytic rules from operating, depending on what you limit.

After you register a server and it completes the initial system metadata synchronization, you should be able to see your server and any recommendations for it in the Insights console in the Red Hat Cloud Portal.

Red Hat Insights currently provides predictive analytics and recommendations for these Red Hat products:

- Red Hat Enterprise Linux 6.4 and later
- Red Hat Virtualization
- Red Hat Satellite 6 and later
- Red Hat OpenShift Container Platform
- Red Hat OpenStack Platform 7 and later
- Red Hat Ansible Automation Platform

Red Hat Insights Architecture Description

When you register a system with Red Hat Insights, it immediately sends metadata about its current configuration to the Red Hat Insights platform. After registration, the system periodically updates the metadata that it provides to Red Hat Insights. The system sends the metadata with TLS encryption to protect it in transit.

The Red Hat Insights platform analyzes the received data, and displays the result on the <https://console.redhat.com/insights> site.



Figure 14.30: Insights high-level architecture

## Install Red Hat Insights Clients

Insights is included with Red Hat Enterprise Linux 9 as part of the subscription. Earlier versions of Red Hat Enterprise Linux servers require installing the `insights-client` package on the system. The `insights-client` package replaced the `redhat-access-insights` package starting with Red Hat Enterprise Linux 7.5. The following section provides a detailed orientation to install the `insights-client` package and to register your system to Red Hat Insights.

The Insights client periodically updates the metadata that is provided to Insights. Use the `insights-client` command to refresh the client's metadata.

```
[root@host ~]# insights-client
Starting to collect Insights data for host.example.com
Uploading Insights data.
Successfully uploaded report from host.example.com to account 1460291.
View details about this system on console.redhat.com:
https://console.redhat.com/insights/inventory/dc480efd-4782-417e-a496-cb33e23642f0
```

## Register a RHEL System with Red Hat Insights

Registering a RHEL server to Red Hat Insights is a quick task.

Interactively register the system with the Red Hat Subscription Management service.

```
[root@host ~]# subscription-manager register --auto-attach
```

Ensure that the `insights-client` package is installed on your system. The package is installed by default on RHEL 8 and later systems.

```
[root@host ~]# dnf install insights-client
```

Use the `insights-client --register` command to register the system with the Insights service and to upload initial system metadata.

```
[root@host ~]# insights-client --register
```



On Red Hat Insights (<https://console.redhat.com/insights>), ensure that you are logged in and that the system is visible under the **Inventory** section of the web UI.

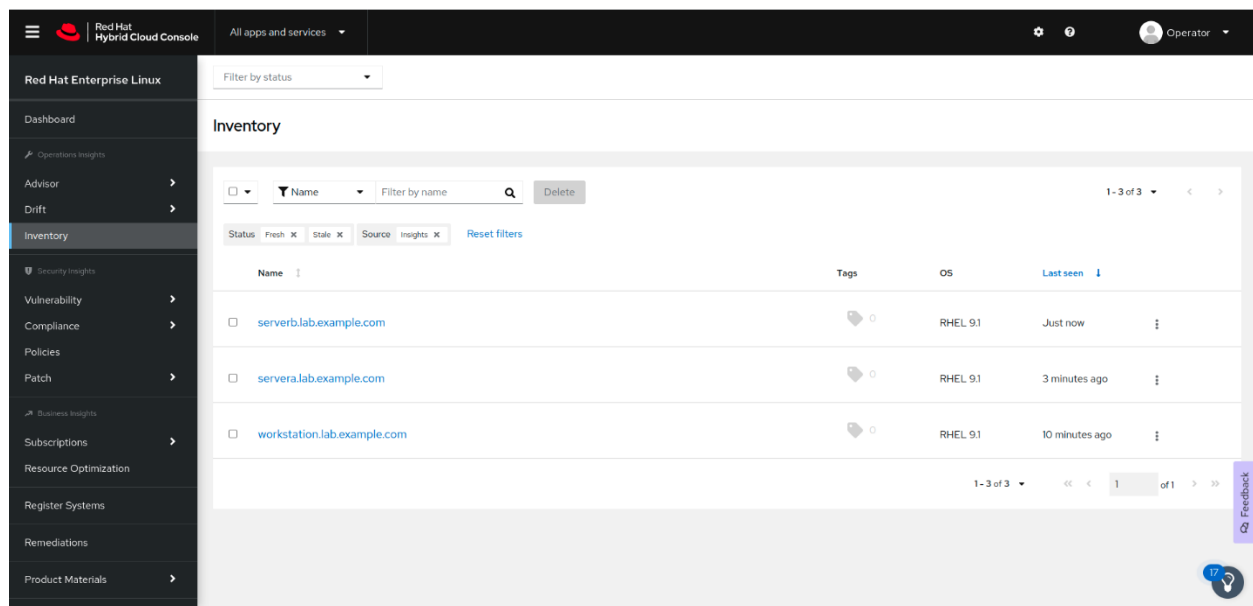


Figure 14.31: Insights inventory on the Cloud Portal

## Red Hat Insights Console Navigation

Red Hat Insights provides a family of services that you access with a web browser at the <https://console.redhat.com/insights> website.

### Detect Configuration Issues with the Advisor Service

The Advisor service reports configuration issues that impact your systems. You can access the service from the **Advisor** → **Recommendations** menu.

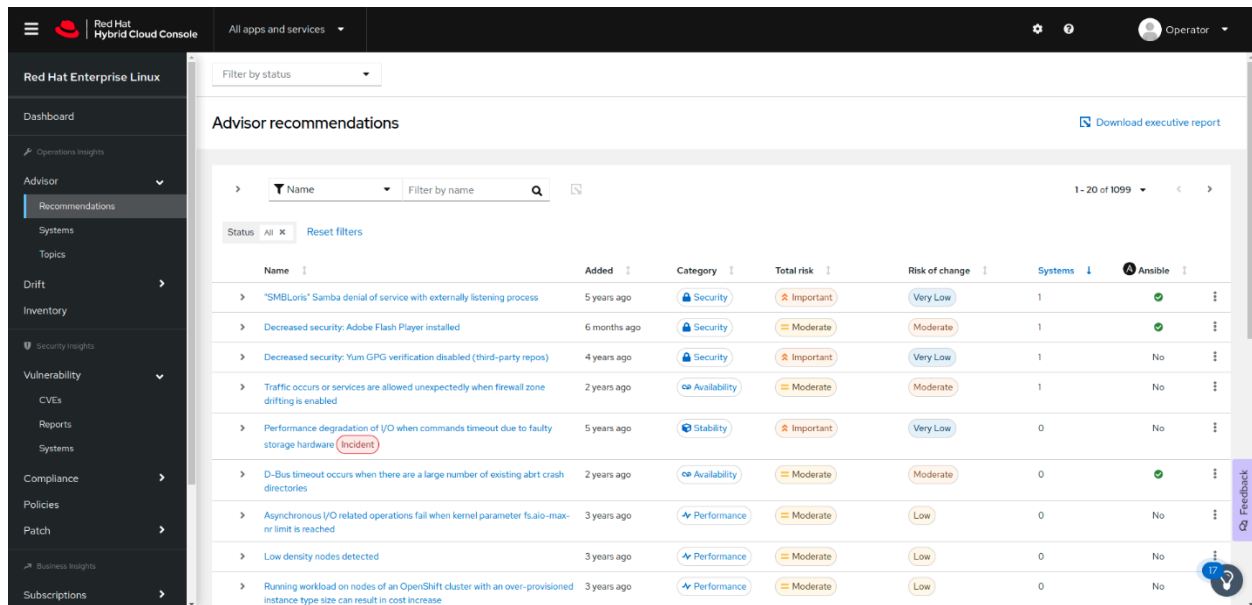


Figure 14.32: Recommendations from the Advisor Service

For each issue, Red Hat Insights provides information to help you to understand the problem, prioritize work to address it, determine what mitigation or remediation is available, and automate resolution with an Ansible Playbook. Red Hat Insights also provides links to Knowledgebase articles on the Customer Portal.

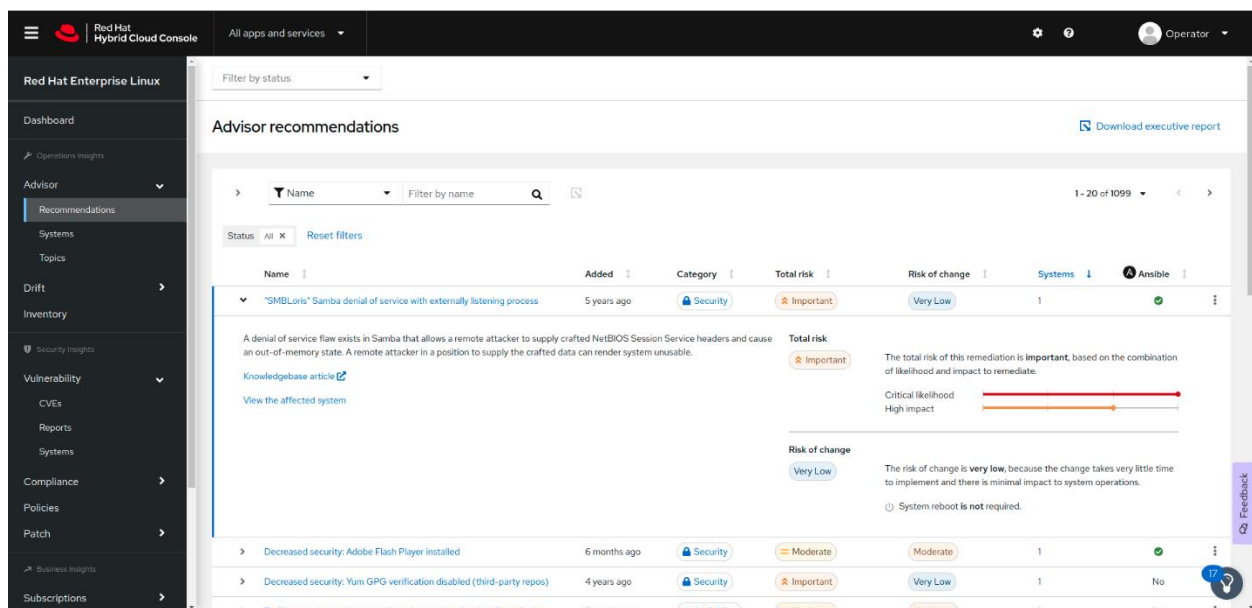


Figure 14.33: Details of an issue

The Advisor service evaluates in two categories the risk that an issue presents to your system:

## Total risk

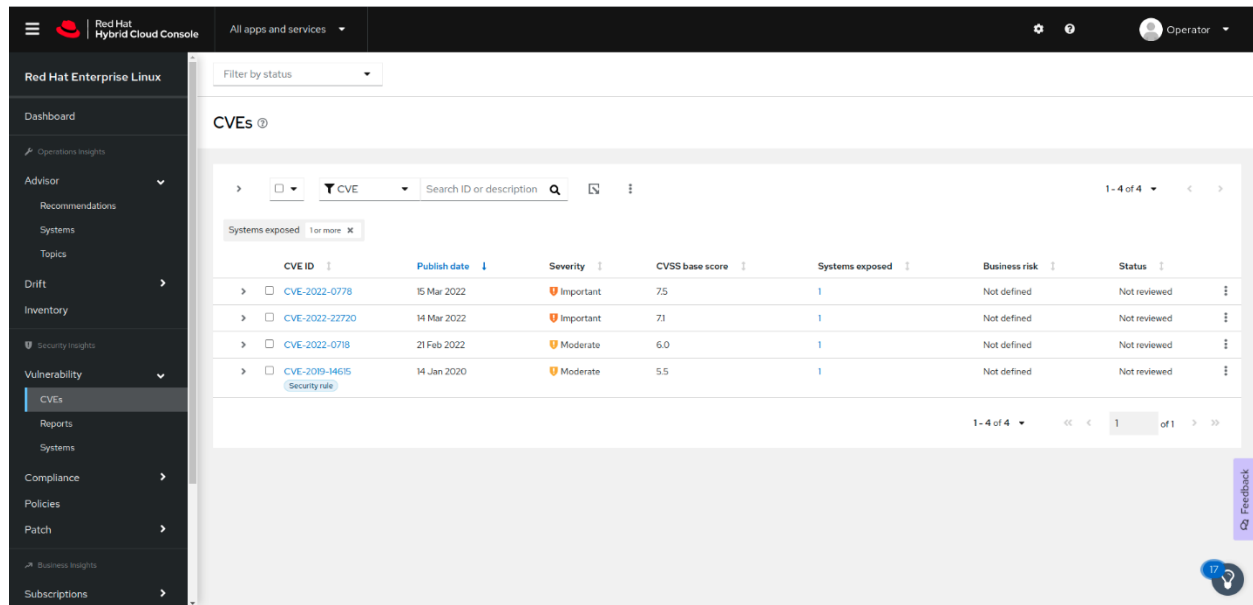
Indicates the impact of the issue on your system.

## Risk of change

Indicates the impact of the remediation action to your system. For example, you might need to restart the system.

## Assess Security with the Vulnerability Service

The Vulnerability service reports common vulnerabilities and exposures (CVEs) that impact your systems. You access the service from the **Vulnerability** → **CVEs** menu.



CVE ID	Publish date	Severity	CVSS base score	Systems exposed	Business risk	Status
<a href="#">CVE-2022-0778</a>	15 Mar 2022	Important	7.5	1	Not defined	Not reviewed
<a href="#">CVE-2022-22720</a>	14 Mar 2022	Important	7.1	1	Not defined	Not reviewed
<a href="#">CVE-2022-0718</a>	21 Feb 2022	Moderate	6.0	1	Not defined	Not reviewed
<a href="#">CVE-2019-14615</a>	14 Jan 2020	Moderate	5.5	1	Not defined	Not reviewed

Figure 14.34: Report from the Vulnerability service

For each CVE, Insights provides additional information and lists the exposed systems. You can click the **Remediate** button to create an Ansible Playbook for remediation.

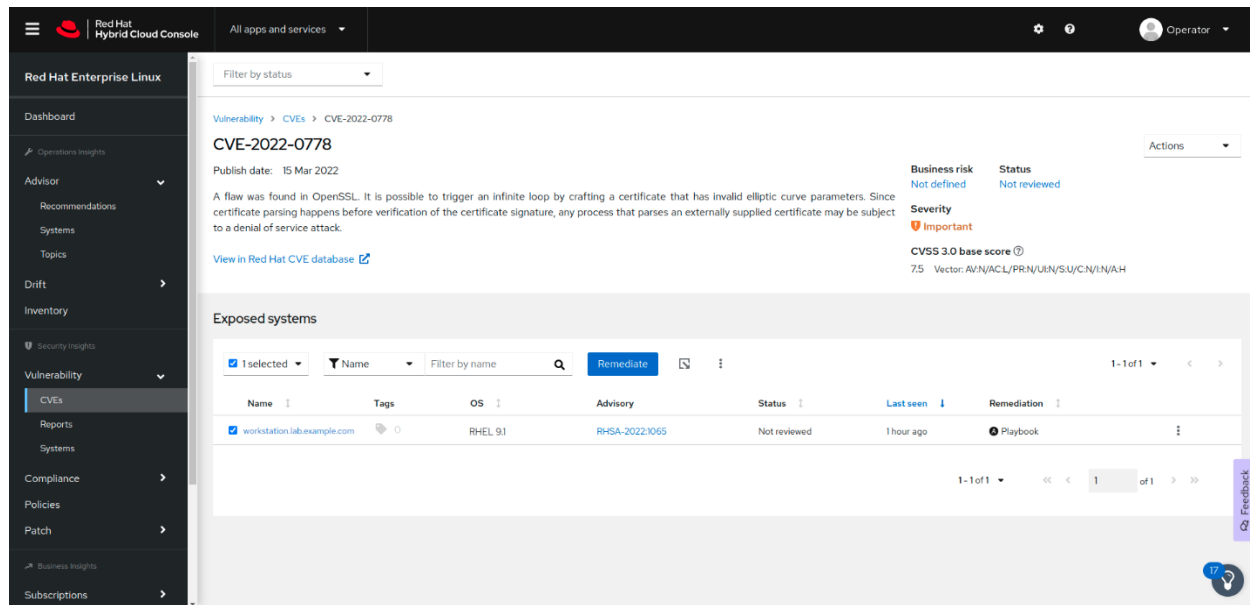


Figure 14.35: Details of a CVE

## Analyze Compliance by Using the Compliance Service

The Compliance service analyzes your systems and reports their compliance level to an OpenSCAP policy. The OpenSCAP project implements tools to verify the compliance of a system against a set of rules. Red Hat Insights provides the rules to evaluate your systems against different policies, such as the Payment Card Industry Data Security Standard (PCI DSS).

## Update Packages with the Patch Service

The Patch service lists the Red Hat Product Advisories that apply to your systems. It can also generate an Ansible Playbook, which you can run to update the relevant RPM packages for the applicable advisories. To access the list of advisories for a specific system, use the **Patch** → **Systems** menu. Click the **Apply all applicable advisories** button to generate the Ansible Playbook for a system.

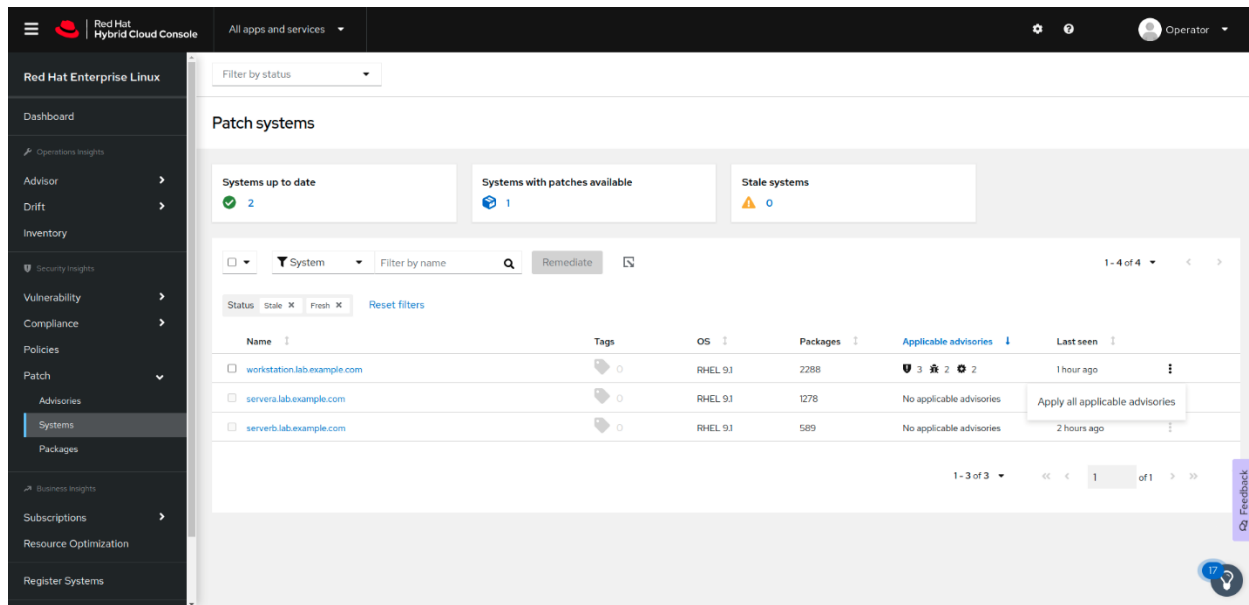


Figure 14.36: Patching a system

## Compare Systems with the Drift Service

With the Drift service, you can compare systems, or obtain a system history. You can use this service for troubleshooting, by comparing a system to a similar system, or to a previous system state. You can access the service from the **Drift** → **Comparison** menu.

The following figure shows that you can use Red Hat Insights to compare the same system at two different times:

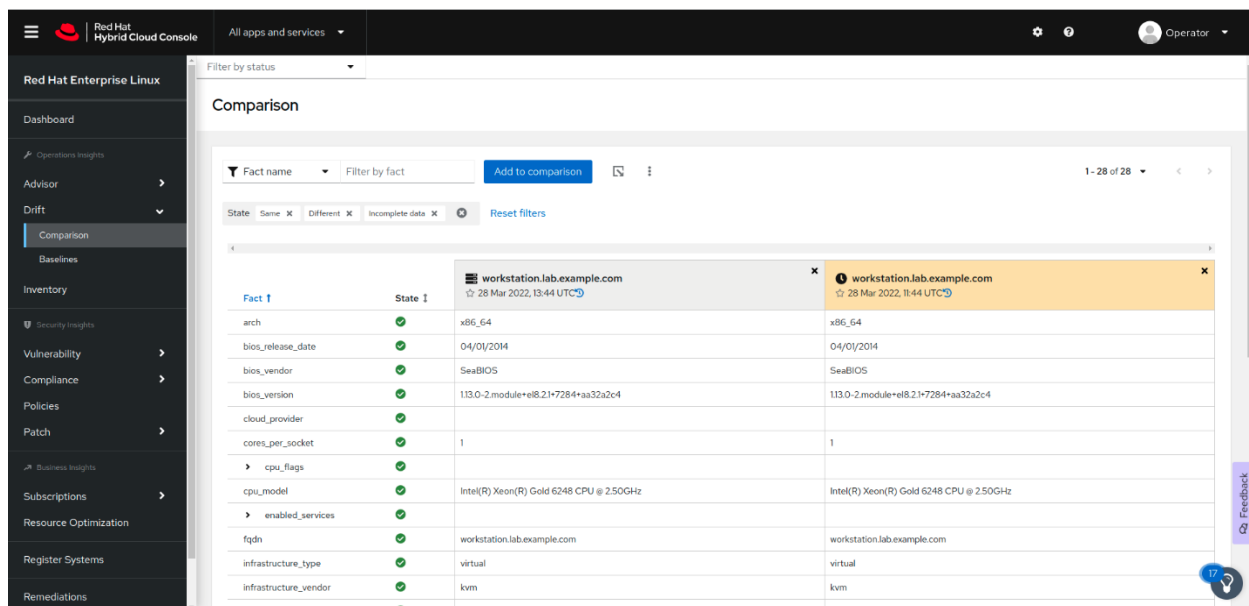


Figure 14.37: Comparing system history

## Trigger Alerts with the Policies Service

By using the Policies service, you can create rules to monitor your systems and send alerts when a system does not comply with your rules. Red Hat Insights evaluates the rules every time that a system synchronizes its metadata. You can access the Policies service from the **Policies** menu.

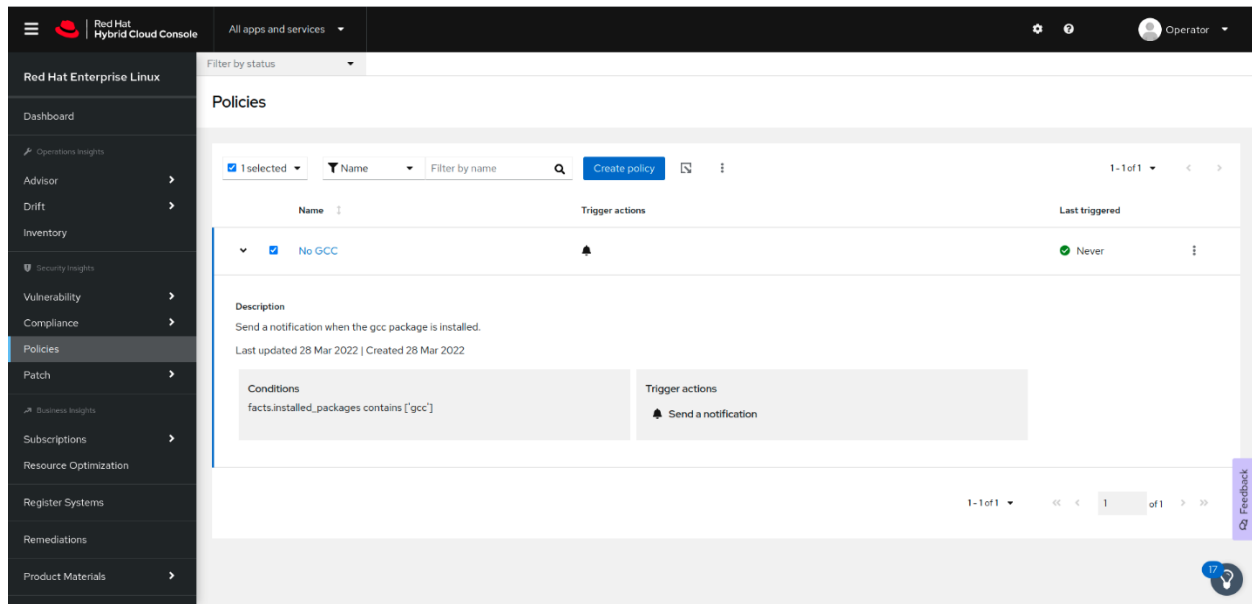


Figure 14.38: Details of a custom rule  
Inventory, Remediation Playbooks, and Subscriptions Monitoring

The **Inventory** page provides a list of the systems that you registered with Red Hat Insights. The **Last seen** column displays the time of the most recent metadata update for each system. By clicking a system name, you can review its details and directly access the Advisor, Vulnerability, Compliance, and Patch services for that system.

The **Remediations** page lists all the Ansible Playbooks that you created for remediation. You can download the playbooks from that page.

By using the **Subscription** page, you can monitor your Red Hat subscription usage.

## References

`insights-client(8)` and `insights-client.conf(5)` man pages

For more information about Red Hat Insights, refer to the *Product Documentation for Red Hat Insights* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_insights](https://access.redhat.com/documentation/en-us/red_hat_insights)

For more information about excluding data that Insights collects, refer to the *Red Hat Insights Client Data Obfuscation* and *Red Hat Insights Client Data Redaction* chapters in the *Client Configuration Guide for Red Hat Insights* at [https://access.redhat.com/documentation/en-us/red\\_hat\\_insights/2021/html-single/client\\_configuration\\_guide\\_for\\_red\\_hat\\_insights/assembly-main-client-cg](https://access.redhat.com/documentation/en-us/red_hat_insights/2021/html-single/client_configuration_guide_for_red_hat_insights/assembly-main-client-cg)

Information about the data that Red Hat Insights collects is available at [System Information Collected by Red Hat Insights](#)

[Previous](#) [Next](#)

## Quiz: Detect and Resolve Issues with Red Hat Insights

Choose the correct answers to the following questions:

1.

2.

1. In which order do the following events occur when managing a Red Hat Enterprise Linux system with Red Hat Insights?

1) Red Hat Insights analyzes system metadata to determine which issues and recommendations apply.

2) The Insights client uploads system metadata to the Red Hat Insights service.

3) The administrator views the recommended actions in the Red Hat Insights customer portal.

4) The Insights client collects system metadata on the Red Hat Enterprise Linux system.

A

1, 2, 3, 4



B

4, 2, 1, 3

C

4, 2, 3, 1

D

4, 1, 2, 3

3. CheckResetShow Solution

4.

5.

2. Which command do you use to register a client to Red Hat Insights?

A

`insights-client --register`

B

`insights-client --no-upload`

C

`subscription-manager register`

D

`insights-client --unregister`

6. CheckResetShow Solution

7.

8.

3. From which page in the Red Hat Insights console can you

generate  
an  
Ansible  
Playbook  
to update  
the RPM  
packages  
on a  
system?

- A                    **Advisor → Recommendations**
- B                    **Vulnerability → Systems**
- C                    **Patch → Systems**
- D                    Remediations

## 9. CheckResetShow Solution

[Previous](#) [Next](#)

## Summary

- The web console is a web-based management interface to your server, and is based on the open source cockpit service.
- The web console provides graphs of system performance, graphical tools to manage system configuration and to inspect logs, and interactive terminal interfaces.
- The Red Hat Customer Portal provides access to documentation, downloads, optimization tools, support case management, and subscription and entitlement management for your Red Hat products.
- The `redhat-support-tool` command-line tool queries Knowledgebase and works with support cases.
- Red Hat Insights is a SaaS-based predictive analytics tool to help you to identify and remediate threats to your systems' security, performance, availability, and stability.

[Previous](#) [Next](#)

# Chapter 15. Comprehensive Review

## Comprehensive Review

### Lab: Manage Files from the Command Line

### Lab: Manage Users and Groups, Permissions, and Processes

### Lab: Configure and Manage a Server

### Lab: Manage Networks

### Lab: Mount File Systems and Find Files

## Abstract

<b>Goal</b>	Review tasks from <i>Red Hat System Administration I</i> .
<b>Objectives</b>	<ul style="list-style-type: none"><li>Review tasks from <i>Red Hat System Administration I</i>.</li></ul>
<b>Sections</b>	<ul style="list-style-type: none"><li>Comprehensive Review</li></ul>
<b>Labs</b>	<ul style="list-style-type: none"><li>Manage Files from the Command Line</li><li>Manage Users and Groups, Permissions, and Processes</li><li>Configure and Manage a Server</li><li>Manage Networks</li><li>Mount File Systems and Find Files</li></ul>

## Comprehensive Review

### Objectives

Demonstrate knowledge and skills learned in *Red Hat System Administration I*.

### Reviewing Red Hat System Administration I

Before beginning the comprehensive review for this course, you should be comfortable with the topics covered in each chapter.

You can refer to earlier sections in the textbook for extra study.

[Chapter 1, Get Started with Red Hat Enterprise Linux](#)

Define open source, Linux, Linux distributions, and Red Hat Enterprise Linux.

Explain the purpose of open source, Linux, Linux distributions, and Red Hat Enterprise Linux.

### [Chapter 2, Access the Command Line](#)

Log in to a Linux system and run simple commands from the shell.

- Log in to a Linux system and run simple commands from the shell.
- Log in to the Linux system with the GNOME desktop environment to run commands from a shell prompt in a terminal program.
- Save time when running commands from a shell prompt with Bash shortcuts.

### [Chapter 3, Manage Files from the Command Line](#)

Copy, move, create, delete, and organize files from the Bash shell.

- Describe how Linux organizes files, and the purposes of various directories in the file-system hierarchy.
- Specify the absolute location and relative location of files to the current working directory, determine and change the working directory, and list the contents of directories.
- Create, copy, move, and remove files and directories.
- Create multiple file name references to the same file with hard links and symbolic (or "soft") links.
- Efficiently run commands that affect many files by using pattern matching features of the Bash shell.

### [Chapter 4, Get Help in Red Hat Enterprise Linux](#)

Resolve problems by using local help systems.

Find information in local Linux system manual pages.

### [Chapter 5, Create, View, and Edit Text Files](#)

Create, view, and edit text files from command output or in a text editor.

- Save output or errors to a file with shell redirection, and process command output through multiple command-line programs with pipes.
- Create and edit text files from the command line with the `vim` editor.

- Set shell variables to run commands, and edit Bash startup scripts to set shell and environment variables to modify the behavior of the shell and programs that are run from the shell.

### [\*Chapter 6, Manage Local Users and Groups\*](#)

Create, manage, and delete local users and groups, and administer local password policies.

- Describe the purpose of users and groups on a Linux system.
- Switch to the superuser account to manage a Linux system, and grant other users superuser access through the `sudo` command.
- Create, manage, and delete local user accounts.
- Create, modify, and delete local group accounts.
- Set a password management policy for users, and manually lock and unlock user accounts.

### [\*Chapter 7, Control Access to Files\*](#)

Set Linux file-system permissions on files and interpret the security effects of different permission settings.

- List file-system permissions on files and directories, and interpret the effects of those permissions on access by users and groups.
- Change the permissions and ownership of files with command-line tools.
- Control the default permissions of user-created files, explain the effects of special permissions, and use special and default permissions to set the group owner of files that are created in a directory.

### [\*Chapter 8, Monitor and Manage Linux Processes\*](#)

Evaluate and control processes that run on a Red Hat Enterprise Linux system.

- Determine status, resource use, and ownership of running programs on a system, to control them.
- Use Bash job control to manage multiple processes that were started from the same terminal session.
- Use commands to kill and communicate with processes, define the characteristics of a daemon process, and stop user sessions and processes.
- Define load average and determine resource-intensive server processes.

## [Chapter 9, Control Services and Daemons](#)

Control and monitor network services and system daemons with the `systemd` service.

- List system daemons and network services that the `systemd` service and socket units started.
- Control system daemons and network services with the `systemctl` command.

## [Chapter 10, Configure and Secure SSH](#)

Configure secure command-line service on remote systems with OpenSSH.

- Log in to a remote system and run commands with `ssh`.
- Configure a user account to use key-based authentication to log in to remote systems securely without a password.
- Disable direct logins as root and password-based authentication for the OpenSSH service.

## [Chapter 11, Manage Networking](#)

Configure network interfaces and settings on Red Hat Enterprise Linux servers.

- Fundamental concepts of network addressing and routing for a server.
- Test and inspect the current network configuration with command-line utilities.
- Manage network settings and devices with the `nmtui` command.
- Modify network configuration by editing configuration files.
- Configure a server's static hostname and its name resolution and test the results.

## [Chapter 12, Install and Update Software Packages](#)

Download, install, update, and manage software packages from Red Hat and DNF package repositories.

- Register a system to your Red Hat account and assign it entitlements for software updates and support services with Red Hat Subscription Management.
- Explain how software is provided as RPM packages, and investigate the DNF and RPM installed system packages.
- Find, install, and update software packages with the `dnf` command.

- Enable and disable server use of Red Hat or third-party DNF repositories.

### [Chapter 13, Access Linux File Systems](#)

Access, inspect, and use existing file systems on storage that is attached to a Linux server.

- Identify a directory in the file-system hierarchy and the device where it is stored.
- Access the contents of file systems by adding and removing file systems in the file-system hierarchy.
- Search for files on mounted file systems with the `find` and `locate` commands.

### [Chapter 14, Analyze Servers and Get Support](#)

Investigate and resolve issues in the web-based management interface, getting support from Red Hat to help solve problems.

- Activate the web console management interface to remotely manage and monitor the performance of a Red Hat Enterprise Linux server.
- Describe and use the Red Hat Customer Portal key resources to find information from Red Hat documentation and the Knowledgebase.
- Use Red Hat Insights to analyze servers for issues, remediate or resolve them, and confirm that the solution worked.

[Next](#)

## Lab: Manage Files from the Command Line

### Note

If you plan to take the RHCSA exam, then use the following approach to maximize the benefit of this Comprehensive Review: attempt each lab without viewing the solution buttons or referring to the course content. Use the grading scripts to gauge your progress as you complete each lab.

In this review, you manage files, redirect a specific set of lines from a text file to another file, and edit the text files.

## Outcomes

- Manage files from the command line.
- Display a specific number of lines from text files and redirect the output to another file.
- Edit text files.

If you did not reset your workstation and server machines at the end of the last chapter, then save any work that you want to keep from earlier exercises on those machines, and reset them now.

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 rhcsa-rh124-review1
```

## Specifications

- Log in to serverb as the student user.
- Create the `/home/student/grading` directory.
- Create three empty files called `grade1`, `grade2`, and `grade3`, in the `/home/student/grading` directory.
- Capture the first five lines of the `/home/student/bin/manage` file in the `/home/student/grading/review.txt` file.
- Append the last three lines of the `/home/student/bin/manage` file to the `/home/student/grading/review.txt` file. Do not overwrite any existing text in the `/home/student/grading/review.txt` file.
- Copy the `/home/student/grading/review.txt` file to the `/home/student/grading/review-copy.txt` file.
- Edit the `/home/student/grading/review-copy.txt` file so that the `Test JJ` line appears twice.
- Edit the `/home/student/grading/review-copy.txt` file to remove the `Test HH` line.
- Edit the `/home/student/grading/review-copy.txt` file so that a line with `A new` line exists between the `Test BB` line and the `Test CC` line.
- Create the `/home/student/hardcopy` hard link to the `/home/student/grading/grade1` file. You must create the hard link after completing the earlier step to create the `/home/student/grading/grade1` file.



- Create the `/home/student/softcopy` symbolic link to the `/home/student/grading/grade2` file.
- Save the output of a command that lists the contents of the `/boot` directory to the `/home/student/grading/longlisting.txt` file. The output should be a "long listing" that includes file permissions, owner and group owner, size, and modification date of each file. The output should omit hidden files.
- Log in to `serverb` as the `student` user.

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

- Create the `/home/student/grading` directory. If the `/home/student` directory is your current directory, then you do not need to specify the absolute path to the `grading` directory when creating it.

```
[student@serverb ~]$ mkdir grading
```

- In the `/home/student/grading` directory, create three empty files called `grade1`, `grade2`, and `grade3`.
  1. Create the empty files called `grade1`, `grade2`, and `grade3` in the `/home/student/grading` directory. Apply the brace expansion shell feature to create all three files with a single `touch` command.

```
[student@serverb ~]$ touch grading/grade{1,2,3}
```

2. Verify that the `grade1`, `grade2`, and `grade3` files exist in the `/home/student/grading` directory.

```
3. [student@serverb ~]$ ls grading/
```

```
grade1  grade2  grade3
```

- Copy the first five lines of the `/home/student/bin/manage` file to the `/home/student/grading/review.txt` file.

1. View the first five lines of the `/home/student/bin/manage` file and redirect the output to the `/home/student/grading/review.txt` file. Use the single redirection symbol (`>`) to overwrite any existing content in the file.

```
[student@serverb ~]$ head -5 bin/manage > grading/review.txt
```

2. Verify that the `/home/student/grading/review.txt` file contains the following text:

3. Test AA
4. Test BB
5. Test CC
6. Test DD

```
Test EE
```

- Append the last three lines of the `/home/student/bin/manage` file to the `/home/student/grading/review.txt` file. Use the double redirection symbol (`>>`) to append the output and preserve the contents of the file.

1. View the last three lines of the `/home/student/bin/manage` file and append the output to the `/home/student/grading/review.txt` file.

```
[student@serverb ~]$ tail -3 bin/manage >> grading/review.txt
```

2. Verify that the `/home/student/grading/review.txt` file contains the following text:

3. Test AA
4. Test BB
5. Test CC
6. Test DD
7. Test EE
8. Test HH
9. Test II

```
Test JJ
```

- Copy the `/home/student/grading/review.txt` file to the `/home/student/grading/review-copy.txt` file.

1. Navigate to the /home/student/grading directory.

```
[student@serverb ~]$ cd grading/
```

```
[student@serverb grading]$
```

3. Copy the /home/student/grading/review.txt file to the /home/student/grading/review-copy.txt file.

```
[student@serverb grading]$ cp review.txt review-copy.txt
```

4. Navigate back to the home directory of the student user.

```
[student@serverb grading]$ cd
```

```
[student@serverb ~]$
```

- Edit the /home/student/grading/review-copy.txt file to have two sequential Test JJ lines.

1. Use the vim text editor to open the /home/student/grading/review-copy.txt file.

```
[student@serverb ~]$ vim grading/review-copy.txt
```

2. From the command mode in vim, scroll down to the Test JJ line. Press the **y** key twice to copy the line of text, and press the **p** key to paste it below the cursor. Type **wq** to save the changes and quit vim. Verify that the /home/student/grading/review-copy.txt file contains the following text:

```
3. Test AA
4. Test BB
5. Test CC
6. Test DD
7. Test EE
8. Test HH
9. Test II
10. Test JJ
```

```
Test JJ
```

- Edit the `/home/student/grading/review-copy.txt` file to remove the `Test HH` line.

1. Use the Vim text editor to open the `/home/student/grading/review-copy.txt` file.

```
[student@serverb ~]$ vim grading/review-copy.txt
```

2. From the command mode in Vim, scroll down to the `Test HH` line. Press the **d** key twice on your keyboard to delete the line of text. Type **wq** to save the changes and quit vim. Verify that the `/home/student/grading/review-copy.txt` file contains the following text:

```
3. Test AA
4. Test BB
5. Test CC
6. Test DD
7. Test EE
8. Test II
9. Test JJ
```

```
Test JJ
```

- Edit the `/home/student/grading/review-copy.txt` file so that the line with `A new line` exists between the `Test BB` line and the `Test CC` line.

1. Use the Vim text editor to open the `/home/student/grading/review-copy.txt` file.

```
[student@serverb ~]$ vim grading/review-copy.txt
```

2. From the command mode in Vim, scroll down to the `Test cc` line. Press the **i** key to switch to the insert mode while keeping the cursor at the beginning of the `Test cc` line. From the insert mode, press **Enter** to create a blank line above the cursor. Use the up arrow to navigate to the blank line and create the line of `A new line` text. Press the **Esc** key to switch back to the command mode. Type **wq** to save the changes and to quit Vim. Verify that the `/home/student/grading/review-copy.txt` file contains the following text.

```
3. Test AA
4. Test BB
5. A new line
```

6. Test CC
7. Test DD
8. Test EE
9. Test II
10. Test JJ

Test JJ

- Create the /home/student/hardcopy hard link to the /home/student/grading/grade1 file.

1. Create the /home/student/hardcopy hard link to the /home/student/grading/grade1 file.

```
[student@serverb ~]$ ln grading/grade1 hardcopy
```

2. View the link count of the /home/student/grading/grade1 file.

3. [student@serverb ~]\$ ls -l grading/grade1

```
-rw-rw-r--. 2 student student 0 Mar  6 16:45 grading/grade1
```

- Create the /home/student/softcopy symbolic link to the /home/student/grading/grade2 file.

1. Create the /home/student/softcopy symbolic link to the /home/student/grading/grade2 file.

```
[student@serverb ~]$ ln -s grading/grade2 softcopy
```

2. View the properties of the /home/student/softcopy symbolic link.

3. [student@serverb ~]\$ ls -l softcopy

```
lrwxrwxrwx. 1 student student 14 Mar  6 17:58 softcopy -> grading/grade2
```

- List the contents of the /boot directory and redirect the output to the /home/student/grading/longlisting.txt file. The output should be a long listing that includes the file permissions, owner and group owner, size, and modification date of each file. The output should omit hidden files.

1. View the contents of the `/boot` directory in the long listing format, and omit hidden files. Redirect the output to the `/home/student/grading/longlisting.txt` file.

```
[student@serverb ~]$ ls -l /boot > grading/longlisting.txt
```

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

```
3. [student@serverb ~]$ exit
```

4. logout

```
Connection to serverb closed.
```

## 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 rhcsa-rh124-review1
```

## 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 rhcsa-rh124-review1
```

This concludes the section.

[Previous](#) [Next](#)

# Lab: Manage Users and Groups, Permissions, and Processes

Note

If you plan to take the RHCSA exam, then use the following approach to maximize the benefit of this Comprehensive Review: attempt each lab without viewing the solution buttons or referring to the course content. Use the grading scripts to gauge your progress as you complete each lab.

In this review, you manage user and group accounts, set permissions on files and directories, and manage processes.

## Outcomes

- Manage user accounts and groups.
- Set permissions on files and directories.
- Identify and manage high CPU-consuming processes.

If you did not reset your workstation and server machines at the end of the last chapter, then save any work that you want to keep from earlier exercises on those machines, and reset them now.

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 rhcsa-rh124-review2
```

## Specifications

- Log in to serverb as the `student` user.
- Identify and terminate the process that currently uses the most CPU time.
- Create the database group with a GID of 50000.
- Create the `dbadmin1` user and configure it with the following requirements:
  - Add the database group as a supplementary group.
  - Set the password to `redhat` and force a password change on first login.
  - Allow the password to change after 10 days since the day of the last password change.
  - Set the password expiration to 30 days since the day of the last password change.
  - Allow the user to use the `sudo` command to run any command as the superuser.

- Configure the default umask as 007 for the dbadmin user.
- Create the /home/dbadmin1/grading/review2 directory with dbadmin1 as the owning user and the database group as the owning group.
- Configure the /home/dbadmin1/grading/review2 directory so that the database group owns any file or sub-directory that is created in this directory, irrespective of which user created the file. Configure the permissions on the directory to allow members of the database group to access the directory and to create contents in it. All other users should have read and execute permissions on the directory.
- Ensure that users are allowed to delete only files that they own from the /home/dbadmin1/grading/review2 directory.
- Log in to serverb as the student user.

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

- Identify and terminate the process that currently uses the most CPU time.
  1. Use the top command to view the real-time system CPU consumption.

```
[student@serverb ~]$ top
```

2. From the interactive interface of the top command, look at the %CPU column and confirm that a dd process is consuming the most CPU resources.

3. ...output omitted...

4. PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
5. <b>2303</b>	student	20	0	217048	944	876	R	<b>99.7</b>	0.1	100:11.64	<b>dd</b>

...output omitted...

The dd process in the preceding output has the 2303 PID. This process is consuming 99.7% of the CPU resources. The PID and the percentage of CPU resource consumption would vary in your system.

6. From the interactive interface of the top command, type **k** to kill the dd process with the 2303 PID, as you determined in the preceding step. After you type **k** in the top command, if the default PID that is shown in the prompt matches the



PID of the process to terminate, then press the **Enter** key. If the suggested PID does not match, then specify the PID interactively.

```
7. ...output omitted...
```

```
8. PID to signal/kill [default pid = 2303] Enter
```

```
...output omitted...
```

9. Use the default SIGTERM signal to terminate the process.

```
10....output omitted...
```

```
11.Send pid 2833 signal [15/sigterm] Enter
```

```
...output omitted...
```

12.Press the **q** key to quit the interactive interface of the top command.

- Create the database group with a GID of 50000.

1. Switch to the root user.

```
2. [student@serverb ~]$ sudo -i
```

```
3. [sudo] password for student: student
```

```
[root@serverb ~]#
```

4. Create the database group with a GID of 50000.

```
[root@serverb ~]# groupadd -g 50000 database
```

- Create the dbadmin1 user. Add the database group as a supplementary group. Set the password to redhat and force a password change on the user's first login. Allow the password to change after 10 days since the day of the last password change. Set the password expiration to 30 days since the day of the last password change. Allow the user to use the sudo command to run any command as the superuser. Configure the default umask as 007.

1. Create the dbadmin1 user. Add the database group as a supplementary group.

```
[root@serverb ~]# useradd -G database dbadmin1
```

2. Set the password of the dbadmin1 user to redhat.

```
3. [root@serverb ~]# passwd dbadmin1
4. Changing password for user dbadmin1.
5. New password: redhat
6. BAD PASSWORD: The password is shorter than 8 characters
7. Retype new password: redhat
```

```
passwd: all authentication tokens updated successfully.
```

8. Force the dbadmin1 user to change its password on the next login.

```
[root@serverb ~]# chage -d 0 dbadmin1
```

9. Set the password's minimum age of the dbadmin1 user to 10 days.

```
[root@serverb ~]# chage -m 10 dbadmin1
```

10. Set the password's maximum age of the dbadmin1 user to 30 days.

```
[root@serverb ~]# chage -M 30 dbadmin1
```

11. Enable the dbadmin1 user to use the sudo command to run any command as the superuser. Use the `vim /etc/sudoers.d/dbadmin1` command to create the file and add the following content:

```
12. [root@serverb ~]# vim /etc/sudoers.d/dbadmin1
```

```
dbadmin1 ALL=(ALL) ALL
```

13. Switch to the dbadmin1 user. Append the `umask 007` line to the `/home/dbadmin1/.bashrc` file.

```
14. [root@serverb ~]# su - dbadmin1
```

```
[dbadmin1@serverb ~]$ echo "umask 007" >> .bashrc
```

15. Source the `~/ .bashrc` file to update the umask.

```
[dbadmin1@serverb ~]$ source ~/.bashrc
```

- Create the /home/dbadmin1/grading/review2 directory with dbadmin1 as the owning user and the database group as the owning group.

1. Use the mkdir command -p option to create the /home/dbadmin1/grading/review2 directory.

```
[dbadmin1@serverb ~]$ mkdir -p /home/dbadmin1/grading/review2
```

2. Recursively set dbadmin1 and database as the respective owning user and group of the /home/dbadmin1/ directory and subdirectories.

```
[dbadmin1@serverb ~]$ chown -R dbadmin1:database /home/dbadmin1/
```

3. Recursively set group execute permissions on the /home/dbadmin1 directory and subdirectories. This permission allow members of the database group to traverse the /home/dbadmin1 directory structure.

```
[dbadmin1@serverb ~]$ chmod -R g+x /home/dbadmin1
```

- Configure the /home/dbadmin1/grading/review2 directory to allow members of the database group to create contents in it. All other users should have read and execute permissions on the directory.

1. Apply the SetGID special permission on the /home/dbadmin1/grading/review2 directory so that the database group owns files that are created in the directory.

```
[dbadmin1@serverb ~]$ chmod g+s /home/dbadmin1/grading/review2
```

2. Apply the 775 permission mode on the /home/dbadmin1/grading/review2 directory.

```
[dbadmin1@serverb ~]$ chmod 775 /home/dbadmin1/grading/review2
```

- Ensure that users are allowed to delete only files that they own from the /home/dbadmin1/grading/review2 directory.

1. Apply the sticky bit special permission on the /home/dbadmin1/grading/review2 directory.

```
[dbadmin1@serverb ~]$ chmod o+t /home/dbadmin1/grading/review2
```

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

```
3. [dbadmin1@serverb ~]$ exit
```

```
4. logout
```

```
5. [root@serverb ~]# exit
```

```
6. logout
```

```
7. [student@serverb ~]$ exit
```

```
8. logout
```

```
Connection to serverb closed.
```

## 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 rhcsa-rh124-review2
```

## 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 rhcsa-rh124-review2
```

This concludes the section.

[Previous](#) [Next](#)

# Lab: Configure and Manage a Server

## Note

If you plan to take the RHCSA exam, then use the following approach to maximize the benefit of this Comprehensive Review: attempt each lab without viewing the

solution buttons or referring to the course content. Use the grading scripts to gauge your progress as you complete each lab.

In this review, you configure, secure, and use the SSH service to access a remote machine, and manage packages with the `dnf` utility.

## Outcomes

- Create a new SSH key pair.
- Disable SSH logins as the root user.
- Disable password-based SSH logins.
- Update the time zone of a server.
- Install packages and package modules by using the `dnf` command.

If you did not reset your workstation and server machines at the end of the last chapter, then save any work that you want to keep from earlier exercises on those machines, and reset them now.

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 rhcsa-rh124-review3
```

## Specifications

- Log in to `serverb` as the `student` user.
- Generate SSH keys for the `student` user. Do not protect the private key with a passphrase. Save the private and public keys as the `/home/student/.ssh/review3_key` and `/home/student/.ssh/review3_key.pub` files respectively.
- Configure the `student` user on `servera` to accept logins that are authenticated by the `review3_key` SSH key pair. The `student` user on `serverb` should be able to log in to `servera` by using SSH without entering a password.
- On `serverb`, configure the `sshd` service to prevent the root user from logging in.
- On `serverb`, configure the `sshd` service to prevent users from using their passwords to log in. Users should still be able to authenticate logins by using an SSH key pair.

- Install the zsh package on the serverb machine.
- Set the time zone of serverb to Asia/Kolkata.
- Log in to serverb as the student user.

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

- Generate SSH keys for the student user. Do not protect the private key with a passphrase. Name the private and public key files /home/student/.ssh/review3\_key and /home/student/.ssh/review3\_key.pub respectively.

```
[student@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa): /home/student/.ssh/review3_key
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/student/.ssh/review3_key.
Your public key has been saved in /home/student/.ssh/review3_key.pub.
The key fingerprint is:
SHA256:Uqefehw+vRfm94fQZDoz/6IfNYSLK/OpiQ4n6lrKIbY student@serverb.lab.example.com
The key's randomart image is:
+---[RSA 3072]-----+
| .+=oBo+          |
| ...O * =         |
| .. + % =         |
| . +.B =.         |
| ...*..o S        |
| E.=. o + .       |
| . = oo o .       |
|   *... .         |
|   .oo.           |
```

```
+-----[SHA256]-----+
```

- Configure the student user on servera to accept logins that are authenticated by the review3\_key SSH key pair. The student user on serverb should be able to log in to servera by using SSH without entering a password.

1. Export the review3\_key public key to servera from serverb.

```
2. [student@serverb ~]$ ssh-copy-id -i .ssh/review3_key.pub student@servera
3. /usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/review3.pub"
4. /usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
   out any that are already installed
5. /usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
   ed now it is to install the new keys
6. student@servera's password: student
7.
8. Number of key(s) added: 1
9.
10. Now try logging into the machine, with:  "ssh 'student@servera'"
```

and check to make sure that only the key(s) you wanted were added.

11. Verify that you can log in to servera from serverb as the student user by using the review3\_key private key without being prompted for the password.

```
12. [student@serverb ~]$ ssh -i .ssh/review3_key student@servera
13. ...output omitted...
```

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

14. Exit from servera.

```
15. [student@servera ~]$ exit
16. logout
17. Connection to servera closed.
```

```
[student@serverb ~]$
```

- On serverb, configure the sshd service to prevent the root user from logging in.

1. Set the PermitRootLogin parameter to no in the /etc/ssh/sshd\_config file. Use the `sudo vim /etc/ssh/sshd_config` command to edit the configuration file.
2. Reload the sshd service.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

- On serverb, configure the sshd service to prevent users from using their passwords to log in. Users should still be able to authenticate logins by using SSH keys.

1. Set the PasswordAuthentication parameter to no in the /etc/ssh/sshd\_config file. Use the `sudo vim /etc/ssh/sshd_config` command to edit the configuration file.
2. Reload the sshd service.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

- Install the zsh package.

```
[student@serverb ~]$ sudo dnf install zsh
...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
    zsh-5.8-9.el9.x86_64
Complete!
```

- Set the time zone of serverb to Asia/Kolkata.

1. Set the time zone of serverb to Asia/Kolkata.

```
[student@serverb ~]$ sudo timedatectl set-timezone Asia/Kolkata
```

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

3. `[student@serverb ~]$ exit`
4. `logout`
5. Connection to serverb closed.



```
[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 rhcsa-rh124-review3
```

## 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 rhcsa-rh124-review3
```

This concludes the section.

[Previous](#) [Next](#)

# Lab: Manage Networks

## Note

If you plan to take the RHCSA exam, then use the following approach to maximize the benefit of this Comprehensive Review: attempt each lab without viewing the solution buttons or referring to the course content. Use the grading scripts to gauge your progress as you complete each lab.

In this review, you configure and test network connectivity.

## Outcomes

- Configure network settings.
- Test network connectivity.
- Set a static hostname.
- Use locally resolvable canonical hostnames to connect to systems.

If you did not reset your workstation and server machines at the end of the last chapter, then save any work that you want to keep from earlier exercises on those machines, and reset them now.

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 rhcsa-rh124-review4
```

## Important

When you use the `ssh` command to adjust networking settings, an incorrect command might hang or lock out your session. You are then disconnected from the server, and thus the server becomes inaccessible. You must adjust the network configuration through the server console, locally or through a remote console.

In this review, open the `serverb` machine console to adjust the networking settings.

## Specifications

- On `serverb`, determine the name of the Ethernet interface and its active connection profile.
- On `serverb`, create and activate a static connection profile for the available Ethernet interface. The static profile statically sets network settings and does not use DHCP. Configure the static profile to use the network settings in the following table:

Parameter	Setting
IPv4 address	172.25.250.111
Netmask	255.255.255.0
Gateway	172.25.250.254
DNS Server	172.25.250.254

- Set the `serverb` hostname to `server-review4.lab4.example.com`.
- On `serverb`, set `client-review4` as the canonical hostname for the `servera` 172.25.250.10 IPv4 address.

- Configure the static connection profile with an additional IPv4 address of 172.25.250.211 with a netmask of 255.255.255.0. Do not remove the existing IPv4 address. Ensure that serverb responds to all addresses when the static connection is active.
- On serverb, restore the original network settings by activating the original network connection profile.
- Use the system console to log in as the student user on the serverb machine, and switch to the root user.

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

- On serverb, determine the Ethernet interface name and the connection profile name that it uses.

1. Display the network connection information.

```
2. [root@serverb ~]# nmcli device status
3. DEVICE    TYPE        STATE        CONNECTION
4. eth0      ethernet    connected    Wired connection 1
```

```
lo          loopback    unmanaged    --
```

In this example, `eth0` is the Ethernet interface name. The connection profile name is `Wired connection 1`. Create the static connection profile for this interface.

### Note

The network interface and connection profile names might differ from the previous output. Use the name that your system shows to replace the `ethx` placeholder name in subsequent steps.

- On serverb, create the static connection profile for the `ethx` interface. Set the network settings statically so that it does not use DHCP. When done, activate that connection profile. Base the settings on the following table:

IPv4 address	172.25.250.111
--------------	----------------

Netmask	255.255.255.0
Gateway	172.25.250.254
DNS server	172.25.250.254

1. Create the static connection profile with the provided network settings.

```
2. [root@serverb ~]# nmcli connection add con-name static type ethernet \
3. ifname ethX ipv4.addresses '172.25.250.111/24' ipv4.gateway '172.25.250.254' \
4. ipv4.dns '172.25.250.254' ipv4.method manual
```

```
Connection 'static' (ac8620e6-b77e-499f-9931-118b8b015807) successfully added.
```

5. Activate the new connection profile.

```
[root@serverb ~]# nmcli connection up static
```

- Set the serverb hostname to server-review4.lab4.example.com. Verify the new hostname.

1. Configure server-review4.lab4.example.com as the new hostname.

```
2. [root@serverb ~]# hostnamectl hostname server-review4.lab4.example.com
3. [root@serverb ~]# hostname
```

```
server-review4.lab4.example.com
```

- On serverb, set client-review4 as the canonical hostname for the servera 172.25.250.10 IPv4 address.

1. Edit the /etc/hosts file and add client-review4 as a name for the 172.25.250.10 IPv4 address.

```
172.25.250.10 client-review4
```

2. Verify that you can reach the servera 172.25.250.10 IPv4 address by using the canonical client-review4 hostname.

```
3. [root@serverb ~]# ping -c2 client-review4
4. PING client-review4 (172.25.250.10) 56(84) bytes of data.
5. 64 bytes from client-review4 (172.25.250.10): icmp_seq=1 ttl=64 time=0.259 ms
6. 64 bytes from client-review4 (172.25.250.10): icmp_seq=2 ttl=64 time=0.391 ms
```

```
7.  
8. --- client-review4 ping statistics ---  
9. 2 packets transmitted, 2 received, 0% packet loss, time 33ms
```

```
rtt min/avg/max/mdev = 0.259/0.325/0.391/0.066 ms
```

- Modify the static connection profile to configure the additional 172.25.250.211 IPv4 address with the 255.255.255.0 netmask. Do not remove the existing IPv4 address. Verify that serverb responds to all addresses when the static connection profile is active.

1. Add the 172.25.250.211 IP address to the static connection.

```
2. [root@serverb ~]# nmcli connection modify static \
```

```
+ipv4.addresses '172.25.250.211/24'
```

3. Activate the new IP address.

```
4. [root@serverb ~]# nmcli connection up static
```

```
...output omitted...
```

5. From workstation, use the ping command to verify that the 172.25.250.211 IPv4 address is reachable.

```
6. [student@workstation ~]$ ping -c2 172.25.250.211  
7. PING 172.25.250.211 (172.25.250.211) 56(84) bytes of data.  
8. 64 bytes from 172.25.250.211: icmp_seq=1 ttl=64 time=0.246 ms  
9. 64 bytes from 172.25.250.211: icmp_seq=2 ttl=64 time=0.296 ms  
10.  
11. --- 172.25.250.211 ping statistics ---  
12. 2 packets transmitted, 2 received, 0% packet loss, time 50ms
```

```
rtt min/avg/max/mdev = 0.246/0.271/0.296/0.025 ms
```

- On serverb, restore the original settings by activating the original network profile.
1. Return to the console and use the nmcli command to activate the original network profile.

```
2. [root@serverb ~]# nmcli connection up "Wired connection 1"
```

```
...output omitted...
```

The original connection profile name might differ on serverb. Replace the name in this solution with the name from your system. Find the profile name with the `nmcli connection show` command.

3. From workstation, log in to serverb as the student user to verify that the original network settings are active.

```
4. [student@workstation ~]$ ssh student@serverb
```

```
5. ...output omitted...
```

```
[student@server-review4 ~]$
```

6. Exit any extra terminals. Return to the workstation system as the student user.

```
7. [student@server-review4 ~]$ exit
```

```
8. logout
```

```
9. Connection to serverb closed.
```

```
[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 rhcsa-rh124-review4
```

## 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 rhcsa-rh124-review4
```

This concludes the section.

## Lab: Mount File Systems and Find Files

### Note

If you plan to take the RHCSA exam, then use the following approach to maximize the benefit of this Comprehensive Review: attempt each lab without viewing the solution buttons or referring to the course content. Use the grading scripts to gauge your progress as you complete each lab.

In this review, you mount a file system and locate files based on different criteria.

### Outcomes

- Mount an existing file system.
- Find files based on their file name, permissions, and size.

If you did not reset your workstation and server machines at the end of the last chapter, then save any work that you want to keep from earlier exercises on those machines, and reset them now.

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 rhcsa-rh124-review5
```

### Specifications

- Log in to the `serverb` machine as the `student` user and switch to the `root` user.
- Identify the unmounted block device that contains an XFS file system on the `serverb` machine. Mount the block device on the `/review5-disk` directory.
- Locate the `review5-path` file. Create the `/review5-disk/review5-path.txt` file that contains a single line with the absolute path to the `review5-path` file.

- Locate all the files that the contractor1 user and the contractor group own. The files must also have the 640 octal permissions. Save the list of these files in the /review5-disk/review5-perms.txt file.
- Locate all files with a size of 100 bytes. Save the absolute paths of these files in the /review5-disk/review5-size.txt file.
- Log in to the serverb 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 ~]#
```

- Identify the unmounted block device that contains an XFS file system on the serverb machine. Mount the block device on the /review5-disk directory.

1. Identify the unmounted block device that contains the xfs file system.

```
2. [root@serverb ~]# lsblk -fs
3. NAME   FSTYPE LABEL UUID                                MOUNTPOINT
4. ...output omitted...
5. vdb1   xfs      7694653c-45f6-4749-bd87-f2f69c37daa7
6. └─vdb
```

...output omitted...

From the preceding output, the vdb1 block device contains the xfs file system, which is not mounted on the system.

7. Create the /review5-disk directory.

```
[root@serverb ~]# mkdir /review5-disk
```

8. Mount the vdb1 block device on the /review5-disk directory.

```
[root@serverb ~]# mount /dev/vdb1 /review5-disk
```

9. Verify that the vdb1 block device is mounted on the /review5-disk directory.



```
10. [root@serverb ~]# df -Th
```

```
11. Filesystem      Type      Size  Used Avail Use% Mounted on
```

```
12. ...output omitted...
```

```
13. /dev/vdb1       xfs       2.0G   47M   2.0G   3% /review5-disk
```

```
...output omitted...
```

- Locate the review5-path file. Save its absolute path in the /review5-disk/review5-path.txt file.

1. Locate the review5-path file. Redirect all error messages to the /dev/null special file.

```
2. [root@serverb ~]# find / -iname review5-path 2>/dev/null
```

```
/var/tmp/review5-path
```

Note the absolute path to the review5-path file from the preceding output.

3. Use the `vim /review5-disk/review5-path.txt` command and save the absolute path to the review5-path file. The following example shows the expected content of the /review5-disk/review5-path.txt file.

```
4. [root@serverb ~]# cat /review5-disk/review5-path.txt
```

```
/var/tmp/review5-path
```

- Locate all files that the contractor1 user and the contractor group own. The files must have 640 octal permissions. Save the absolute paths of these files in the /review5-disk/review5-perms.txt file.

1. Locate all the files that the contractor1 user and the contractor group own and that have 640 octal permissions. Redirect all the errors to the /dev/null special file.

```
2. [root@serverb ~]# find / -user contractor1 \
```

```
3. -group contractor -perm 640 2>/dev/null
```

```
/usr/share/review5-perms
```

Only the `/usr/share/review5-perms` file meets the criteria of the preceding `find` command. Note the absolute path to the `review5-perms` file.

4. Use the `vim /review5-disk/review5-perms.txt` command and save the absolute path of the `review5-perms` file. The following example shows the expected content of the `/review5-disk/review5-perms.txt` file.

```
5. [root@serverb ~]# cat /review5-disk/review5-perms.txt
```

```
/usr/share/review5-perms
```

- Locate all the files with a size of 100 bytes. Save the absolute paths of these files in the `/review5-disk/review5-size.txt` file.

1. Locate all the files with a size of exactly 100 bytes. Redirect all the errors to the `/dev/null` special file.

```
2. [root@serverb ~]# find / -size 100c 2>/dev/null
```

```
3. /usr/share/licenses/ethtool/LICENSE
```

```
4. /usr/share/doc/libuser
```

```
5. /usr/share/doc/plymouth/AUTHORS
```

```
6. ...output omitted...
```

```
7. /opt/review5-size
```

```
...output omitted...
```

The preceding output might vary depending on the number of files that match the size criteria in your system. Note the absolute paths to all the files from the preceding output.

8. Use the `vim /review5-disk/review5-size.txt` command and save the absolute path of the files from the preceding output. The following example shows the expected content of the `/review5-disk/review5-size.txt` file.

```
9. [root@serverb ~]# cat /review5-disk/review5-size.txt
```

```
10. ...output omitted...
```

```
11. /opt/review5-size
```

```
...output omitted...
```

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

```
13. [root@serverb ~]# exit
14. logout
15. [student@serverb ~]$ exit
16. logout
17. Connection to serverb closed.
```

```
[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 rhcsa-rh124-review5
```

## 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 rhcsa-rh124-review5
```

This concludes the section.

[Previous](#)