Start Lab 01:00:00

Exemplar: Get help in the command line

Lab 1 hour No cost Introductory

This lab may incorporate AI tools to support your learning.

Lab instructions and tasks
Activity overview
Scenario
Start your lab
Task 1. Learn more about commands
Task 2. Explore the useradd command
Task 3. Explore the rm and rmdir commands
Task 4. Determine which command to use
Conclusion
End your lab

Activity overview

As a security analyst, you won't have all the answers all the time, but you can learn where to find them. One of the great things about Linux is that you can get help right through the command line.

In this lab activity, you'll use the `man` and `whatis` commands to get information on other commands and how they work. You'll also use the `apropos` command to search the manual page for a command with a specified string.

When working as a security analyst, you'll likely find it useful to know how to discover which command to use or information about what commands do.

With that in mind, let's explore your scenario.

Scenario

In this scenario, you have to find more information about commands that you need to use. You also need to discover which command to use to perform a certain task.

Here's how you'll do this task: **First**, you'll explore a few commands you can use in the shell to learn more about other commands. **Next**, you'll find an option you need to add to a command. **Third**, you'll use a command to get a brief description of commands so you can identify their differences. **Finally**, you'll identify the command you need to perform a task.

It's time to get ready to explore some of the Linux tools resources!

Disclaimer: For optimal performance and compatibility, it is recommended to use either **Google Chrome** or **Mozilla Firefox** browsers while accessing the labs.

Start your lab

You'll need to start the lab before you can access the materials. To do this, click the green "Start Lab" button at the top of the screen.

Start Lab

After you click the **Start Lab** button, you will see a shell, where you will be performing further steps in the lab. You should have a shell like this:

```
student@linuxlab:~$
```

When you have completed all the tasks, refer to the End your Lab section that follows the tasks for information on how to end your lab.

Task 1. Learn more about commands

In this task, you need to explore a few commands you can use in the shell to learn more about the functionality of other commands.

First, imagine you can't quite remember what the `cat` command does and want a quick reminder.

1. Use the `whatis` command to get a short description of `cat`.

The command to complete this step:

```
whatis cat
```

What are the first two words of the short description of cat returned by whatis?

- the cat
- file concatenator
- concatenate files
- cat is

Submit

Answer: The first two words of the short description returned are "concatenate files".

Next, imagine that you want more details about `cat` and all of its options.

2. Use the `man` command to get more details about `cat`.

The command to complete this step:

```
man cat
```

The `man` command returns a general description of `cat` and information about each of its options:

```
CAT(1)                                                    User Commands

NAME
       cat - concatenate files and print on the standard output

SYNOPSIS
       cat [OPTION]... [FILE]...

DESCRIPTION
       Concatenate FILE(s) to standard output.

       With no FILE, or when FILE is -, read standard input.

       -A, --show-all
           equivalent to -vET

       -b, --number-nonblank
           number nonempty output lines, overrides -n

       -e  equivalent to -vE

  --More--
```

When the first page of information returned by `man` is displayed, the output pauses.

Note: *You can output more information one line at a time by pressing the* ***ENTER*** *key or output the next page of the manual by pressing the space-bar.*

What option can you use to number the output lines of the cat command?

- `none` - it is the default option
- `-e` -enumerate
- `-n` -number
- `-b` -number-nonblank

Submit

Answer: The `-n, --number` option numbers all the output lines.

3. Press **Q** to exit this manual page.

Now, imagine you've remembered there's a command that prints just the first part of a file, but you can't remember the exact command. The `apropos` command is useful in these instances. You can use keywords with `apropos` to find a command.

4. Use `apropos` to find a command that returns the first part of a file:

```
apropos -a first part file
```

Note: *There is no right and wrong when using* `apropos` *in terms of keywords. Think of it as a very focused search. It will only return commands that correspond to keywords you supply. Keep trying if the first returned command does not provide what you need. Also keep in mind that using the* `-a` *option will limit results to only those commands that match all keywords supplied.*

Which command returns the first part of a file?

- `cat`
- `head`
- `tail`
- `tee`

Submit

Answer: The `head` command returns only the first part of a file.

Click **Check my progress** to verify that you have completed this task correctly.

Learn more about commands

Check my progress

Task 2. Explore the useradd command

In this task, imagine that you want to set the expiration date for a temporary user account. You know that you need to use the `useradd` command for this, but you're not quite sure how to complete the task. You realize it might involve adding an option to the command.

1. Use the most appropriate Linux command to get help on the `useradd` command and learn more about all of its options.

The command to complete this step:

```
man useradd
```

Note: *You can output more information one line at a time by pressing the* ***ENTER*** *key or output the next page of the manual by pressing the space-bar.*

Which option can be used with the useradd command to set an expiration date for a temporary user account?

- `-s`
- `-e`
- `-f`
- `-d`

Submit

Answer: The `-e` option can be used to set an expiration date for a temporary user account.

2. Press **Q** to exit this manual page.

Click **Check my progress** to verify that you have completed this task correctly.

Explore the useradd command

Check my progress

Task 3. Explore the rm and rmdir commands

In this task, you need to determine the difference between the `rm` and `rmdir` commands.

Imagine that you've used these commands before, but you can't remember how they're different.

- Use the most appropriate Linux command to quickly remind yourself what each command does.

Note: *This task will require entering two commands, one with* `rm` *and one with* `rmdir`*.*

The commands to complete this step:

```
whatis rm
```

```
whatis rmdir
```

Which of these commands removes only empty directories?

- `rmdir`
- `rm`

Submit

Answer: The `rmdir` command removes only empty directories.

Click **Check my progress** to verify that you have completed this task correctly.

Explore the rm and rmdir commands

Check my progress

Task 4. Determine which command to use

In this task, imagine that you need to create a new group but don't remember what command to use. You need to identify a command that will do this by searching for it through keywords. In this case, use the keywords `create new group`.

- Use the most appropriate Linux command with these keywords to identify what command to use.

The correct command to solve this step:

```
apropos -a create new group
```

What command can you use to create a new group?

- `addnewgroup`
- `setnid`
- `groupadd`
- `newgroup`

Submit

Answer: The `groupadd` can be used to create a new group.

Click **Check my progress** to verify that you have completed this task correctly.

Determine which command to use

Check my progress

Conclusion

Great work!

You now have practical experience in using basic Linux Bash shell commands to

- get a short description of a command,
- display the `man` pages for a command, and
- find commands based on keywords about their function.

This ability will be valuable as you navigate the Linux command line.

End your lab

Before you end the lab, make sure you're satisfied that you've completed all the tasks, and follow these steps:

1. Click **End Lab**. A pop-up box will appear. Click **Submit** to confirm that you're done. Ending the lab will remove your access to the Bash shell. You won't be able to access the work you've completed in it again.
2. Another pop-up box will ask you to rate the lab and provide feedback comments. You can complete this if you choose to.
3. Close the browser tab containing the lab to return to your course.
4. Refresh the browser tab for the course to mark the lab as complete.