

# Create and modify directories and files

Let's make some branches!

What do I mean by that?

Well, in a previous video, we discussed root directories and how other subdirectories branch off of the root directory.

Let's think again about the file directory system as a tree.

The subdirectories are the branches of the tree.

They're all connected from the same root but can grow to make a complex tree.

In this video, we'll create directories and files and learn how to modify them.

When it comes to working with data in security, organization is key.

If we know where information is located, it makes it easier to detect issues and keep information safe.

In a previous video, we've already discussed navigating between directories, but let's take a moment to examine directories more closely.

It's possible you're familiar with the concept of folders for organizing information.

In Linux, we have directories.

Directories help organize files and subdirectories.

For example, within a directory for reports, an analyst may need to create two subdirectories: one for drafts and one for final reports.

Now that we know why we need directories, let's take a look at some essential Linux commands for managing directories and files.

First, let's take note of commands for creating and removing directories.

The `mkdir` command creates a new directory.

In contrast, `rmdir` removes or deletes a directory.

A helpful feature of this command is its built-in warning that lets you know a directory is not empty. This saves you from accidentally deleting files.

Next, you'll use other commands for creating and removing files.

The `touch` command creates a new file, and then the `rm` command removes or deletes a file.

And last, we have our commands for copying and moving files or directories.

The `mv` command moves a file or directory to new location, and `cp` copies a file or directory into a new location.

Now, we're ready to try out these commands.

First, let's use the `pwd` command, and then let's display the names of the files and directories in the `analyst` directory with the `ls` command.

Imagine that we no longer need the `oldreports` directory that appears among the file contents.

Let's take a look at how to remove it.

We input the `rmdir` command and follow it with the name of the directory we want to remove: `oldreports`.

We can use the `ls` command to confirm that `oldreports` has been deleted and no longer appears among the contents.

Now, let's make another change.

We want a new directory for drafts of reports.

We need to use the command: `mkdir` and specify a name for this directory: `drafts`.

If we input `ls` again, we'll notice the new directory `drafts` included among the contents of the `analyst` directory.

Let's change into this new directory by entering: `cd drafts`.

If we run `ls`, it doesn't return any output, indicating that this directory is currently empty.

But next, we'll add some files to it.

Let's say we want to draft new reports on recently installed email and OS patches.

To create these files, we input: `touch email_patches.txt`

and then: `touch OS_patches.txt`.

Running `ls` indicates that these files are now in the `drafts` directory.

What if we realize that we only need a new report on OS patches and we want to delete the email patches report?

To do this, we input the `rm` command and specify the file to delete as: `email_patches.txt`.

Running `ls` confirms that it's been deleted.

Now, let's focus on our commands for moving and copying.

We realized that we have a file called `email policy` in the `reports` folder that is currently in draft format.

We want to move it into the newly created `drafts` folder.

To do this, we need to change into the directory that currently has that file.

Running `ls` in that directory indicates that it contains several files, including `email_policy.txt`.

Then to move that file, we'll enter the `mv` command followed by two arguments.

The first argument after `mv` identifies the file to be moved.

The second argument indicates where to move it.

If we change directories into `drafts` and then display its contents, we'll notice that the `email policy` file has been moved to this directory.

We'll change back into `reports`.

Displaying the file contents confirms that `email_policy` is no longer there.

Okay, one more thing. `vulnerabilities.txt` is a file that we want to keep in the `reports` directory.

But since it affects an upcoming project, we also want to copy it into the project's directory.

Since we're already in the directory that has this file, we'll use the `cp` command to copy it into the `projects` directory.

Notice that the first argument indicates which file to copy, and the second argument provides the path to the directory that it will be copied into.

When we press `Enter`, this copies the `vulnerabilities` file into the `projects` directory while also leaving the original within `reports`.

Isn't it cool what we can do with these commands?

Now, let's focus on one more concept related to modifying files.

In addition to using commands, you can also use applications to help you edit files.

As a security analyst, file editors are often necessary for your daily tasks, like writing or editing reports.

A popular file editor is nano.

It's good for beginners.

You can access this tool through the nano command.

Let's get familiar with nano together.

We'll add a title to our new draft report: OS\_patches.txt.

First, we change into the directory containing that file,

then we input nano followed by the name of the file we want to edit: OS\_patches.txt.

This brings up the nano file editor with that file open.

For now, we'll just enter the title OS Patches by typing this into the editor.

We need to save this before returning to the command line, and to do so, we press Ctrl+O and then enter to save it with the current file name.

Then to exit, we press Ctrl+X.

Great work!

We've covered a lot of topics here—from creating and removing directories and files to copying or moving them, and just now, we've added editing files.

You're well on your way to learning Linux commands!

---

Revision #1

Created 2023-07-05 13:31:33 UTC by naruzkurai

Updated 2023-07-05 13:34:37 UTC by naruzkurai