2.1.4 Enter Shell Commands

Click one of the buttons to take you to that part of the video.

Enter Shell Commands 0:00-0:21

In this demonstration, we're going to spend some time learning how to enter commands at the Linux shell prompt. The Linux shell works in much the same way as the command prompt in a Windows environment. If I want to run a command, all I have to do is type its name here, at the shell prompt.

The pwd Command 0:22-0:56

For example, if I wanted to run the pwd command, I would type the name of the command and then press Enter. pwd command stands for print working directory. This command tells us where we're at, currently, in the Linux file system. If I press Enter, it tells me that I am currently in the /home/rtracy directory within the Linux file system. I do need to point out that with most Linux distributions, the name of the current directory is actually included within the shell prompt itself. And if it's not, you can actually go in and customize it to include it within the shell prompt.

Tilde (~) Character 0:57-1:47

Notice, here, that here that the pwd command says that I am in /home/rtracy, but my shell prompt simply shows a ~ where the name of the directory should go. That's because the ~ on a Linux system is a shortcut that points to the /home directory of whatever user is currently logged in. I'm currently logged in as the rtracy user; therefore, my /home directory is /home/rtracy, and that's what the ~ character points to.

If I were to log in as a different user--let's say I logged into the system as the ksanders user--then the ~ character would actually point to /home/ksanders instead of /home/rtracy. It's just a nice little shortcut that you can use to get around in the file system. If you need to switch back to your /home directory instead of having to type out the entire path to your /home directory, you just type ~.

Full Directory Path 1:48-2:25

No matter what directory you're currently in, in the file system, the pwd command will display the name of that directory along with the full path to that directory. For example, I'm going to type the cd command to change directory, and I'm going to change to the /tmp directory within the Linux file system. Press Enter. Now, if I type 'pwd', it tells me that I am in the /tmp directory. If I want to switch back to my /home directory, I can use that ~ character. I type the cd command again to change directory. And then, instead of typing '/home/rtracy', I just simply type '~'. Hit Enter, and I am back to my rtracy user's /home directory.

The whoami Command 2:26-3:07

Another useful command is the whoami command. The whoami command simply displays the name of the current user--who you're currently logged in as. So, I press Enter, and you can see that I am logged in as rtracy. You might be asking, "Well, what's the significance of that command? Why would you need that?"

Well, on Linux, you will, many times, switch between user accounts. For example, I may log in to the system with my rtracy user, but then when I need to perform administrative tasks, I will need to switch to my superuser account, which, on Linux, is the root user. And by switching back and forth between rtracy and the root user account, I might get confused as to who I am currently logged in as, in which case, I can use the whoami command to determine which user account I'm currently logged in to the system as.

The su Command 3:08-4:01

In order to switch between user accounts, I use the su command. su stands for switch user, and it can be used to switch between different user accounts on the system. For example, I'm currently logged in as rtracy. But if I wanted to switch to the ksanders user account on the system, I would type 'su', space, and then the name of that user--in this case, 'ksanders'.

That actually isn't used very often in my experience. I don't commonly switch between two different standard user accounts. But what I do use su all the time for is to switch to my root user account--my superuser account--so I can perform administrative tasks. In fact, I daresay that is what su is used most for. In fact, if you don't provide a username with the su command, it will just assume that you want to switch to the root user account. So, in order to switch to root, I don't actually have to type out su root; I can just type 'su', and it will say, "Oh, the user wants to switch to root."

Dash (-) Argument 4:02-5:08

There is one thing I need to point out here, and that is the significance of the dash (-) when used with the su command. When you type su and then -, it says, "Switch to this particular user account and load that user's environment variables." So, that that user's path, shell environment, variables, etc., are automatically loaded, and that's a very useful thing to do. In fact, when you use su to switch to the root user account, you'll almost always use - with it to load the root user's environment variables. If you didn't, you'd have a hard time running many different administrative commands on the system.

Let's enter 'su -'. Because I'm switching from the rtracy user account to the root user account, I do have to provide the root user's password. And I'm now logged in as root. Notice, over here, that the shell prompt on this system is configured to automatically display who I'm currently logged in as. Previously, I was logged in as rtracy. Now, I'm logged in as root. Not all shell environments will have that set up by default. But if it's not, then you can, again, use the whoami command to determine who you're currently logged in as.

The exit Command 5:09-5:45

Once I'm logged in as root, I can perform administrative tasks on the system as the superuser account. When I'm done, I should exit out of the root user account and go back to a standard user account for security reasons. So, if I got up and went to the break room and left my computer logged in, somebody couldn't walk in and start doing all kind of nefarious things as my root user--that would be bad. I have two different options for switching back from root to the rtracy user account. One would be to type 'su rtracy' to switch back to the root user account. But an easier way to do it is to simply type 'exit'. When I do, I'm logged out of my root user account, and I'm switched back to my rtracy user account.

The uname Command 5:46-7:13

Another command that can be useful is the uname command. The uname command displays information about the system. If you just run uname without providing any parameters, all it does is tell you the name of the kernel running on this system, which is Linux, and we already knew that. However, if you run uname with the -a option, it provides a little more extensive information about the system itself. Notice that the output is quite a bit longer. The first option displayed right here is the name of the kernel currently running, and we saw that up here. It's the Linux kernel. The next field in the output of the uname command is the hostname of the system I'm using, fs1.corpnet.com.

The next field, right here, is the kernel release number. The next field is the kernel version number along with the version date. The next field is the machine hardware name. This is a 64-bit system; the machine name is x86_64. The next field in the output is the type of CPU in the system, the processor; it's a 64-bit processor. And then the next field specifies the architecture of the operating system. In this case, it's a 64-bit operating system. If we were using a 32-bit operating system, it would show i386 in this field in the output. The last field in the output specifies the name of the operating system. In this case, we're running GNU/Linux.

Summary 7:14-7:16

That's it for this demonstration. In this demo, we learned how to enter commands at the Linux shell prompt.
