4.5.1 System Shutdown

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

System Shutdown 0:00-1:36

As with any other operating system, you need to shut down a Linux system properly. This is called a clean shutdown. This is important because a clean shutdown ensures that any pending disk write operations that haven't been completed yet are actually committed to disk before the system powers off.

In this lesson, we're going to talk about how to do this. As I said, this is very important because if you power off a Linux system uncleanly, you're going to introduce corruption and inconsistencies into your Linux file system. Ask me how I know this--I've done it. Therefore, you need to make sure you always use the proper shutdown procedure. For example, you could push the power button once on the front of the system. This will initiate a clean shutdown. You could also use the GUI in order to shut down the system, or you can use shell commands to shut down the system.

There are some ways that you can actually uncleanly shut down the Linux system. You should not do any of these. For example, you could push and hold the power button on the front of the system. That's a hard shutdown. It does not give the operating system time to write pending disk writes to disk. You could pull the plug from the wall socket, or you could flip the switch on the back of the power supply to shut it off. These are all very, very bad. It's very likely, if you do this, that you're going to introduce errors into the file system, especially if you do it a lot.

Will you have to do these on occasion? Yeah, but they should be saved as an absolute last resort, such as if your system is completely hung and it won't respond at all (which, frankly, doesn't really happen all that often with Linux).

Use a UPS 1:37-2:22

Before we go any further, I strongly recommend that you implement an uninterruptable power supply with your Linux systems. They're not very expensive, and they will save your hide if the power goes off.

This is an example of a UPS here. The UPS itself is plugged into your wall outlet. The UPS has many batteries inside of it that are charged by the wall outlet. That way, if the power goes off over here for some reason, your system can stay running for a period of time off of these batteries. There's usually a data cable that connects to the system. And if the UPS loses power from the wall outlet, it sends a shutdown command to the system so that it goes ahead and shuts down, and you don't risk corrupting files.

Shut Down Using Runlevels 2:23-3:35

In this lesson, we're going to focus on the shell commands that you can use to cleanly shut down a Linux system. For example, if you're working on an older Linux system that's running the init daemon, all you have to do to shut down the system is switch runlevels.

For example, if you want to turn the system all the way off, you can use the 'init 0' command to switch to runlevel zero, which halts the system. Likewise, if you want to reboot the system, you can use the 'init 6' command, and that will switch to runlevel six, which basically reboots the system.

Here's the interesting thing. If you're running a newer Linux distribution that uses systemd, and not init, you can actually still use both of these commands: 'init 6' or 'init 0'. All the shell does is call the corresponding system control command to set the system to the appropriate boot target.

For example, if I were to run init 0, then the shell would just run the systemctl command and set the current boot target to runlevel0.target or poweroff.target. They both do the same thing.

Likewise, if we ran init 6 on the system, the systemctl command would be run again and set the current boot target to runlevel6.target or reboot.target.

Shut Down Using Boot Targets 3:36-4:09

Of course, if you're using a systemd system, you can just use system control manually to specify one of these boot targets. For example, to shut down the system, we would enter 'systemctl isolate poweroff.target'. We set the boot target to poweroff, which is the equivalent of runlevel zero, and you can even use the 'unlevel0.target and accomplish the same thing with the systemctl command. Also, to reboot the system, we can us the isolate action to reset the current boot target to reboot.target. Likewise, you could set the current boot target to 'runlevel6.target', as well, to reboot the system.

Shut Down Using Shell Commands 4:10-4:38

In addition to changing the runlevel or the boot target, most Linux distributions also include other shell commands that you can use to either shut down or reboot the system.

For example, you can use the halt command, which shuts down the system, or you can use the reboot command, which reboots the system. I believe both of these commands have to be run as the root user. You can try it on your particular distribution and see if a standard user can run them, but I'm pretty sure you have to have root-level privileges in order to run these commands.

Use the shutdown Command 4:39-6:46

In addition to halt and reboot, you can also use the shutdown command to either shut down or reboot the system. Halt and reboot work just fine, but shutdown, frankly, works better because it is more flexible than either of these two commands. It has several key advantages. First of all, you can specify that the system go down after a certain period of time, which gives your users time to save their work and log out before the system goes off.

It also allows you to shut down the system at a specified time, even if you're not there to do it, like 3:00 a.m., or something like that. Who wants to go into the office at 3:00 a.m. to shut down a computer?

It also allows you to send a message to all of your logged-in users warning them that a shutdown is about to occur and that they need to log out and save their files.

Also, the shutdown command, after you issue it, will prevent users from logging in while it's waiting for the pending shutdown to occur.

The syntax for using shutdown is shown here. We run 'shutdown', and then we specify the '+m' option to specify the amount of time in minutes to wait before shutting down the system. If you needed to, you could also specify a value of —'now' if you just want the shutdown to occur immediately. If you needed the system to be shut down at a specific time, instead of sitting and calculating how many minutes away that is, you can replace the plus 'm' parameter with just the time that you want the system to go down, such as 0100, to shut the system down at 1:00 in the morning. It uses military time, by the way.

You can also use the -h option, which, as noted here, halts the system, or you can use the dash -r option, which reboots the system. You can also optionally include a message that will inform the users that a shutdown is upcoming and that they need to save all their files and log out.

I also need to point out that there is an option called -c. If you scheduled a pending shutdown with the shutdown command, and then you say, "Oh, that's not a good idea. I need to not shut that system down." You can run 'shutdown -c' to cancel the pending shutdown, and the system will stay running.

Send Messages Using the wall Command 6:47-7:21

You can also use the wall command to send messages to users to inform them of system events, such as a pending shutdown or reboot. To use the wall command, you first create the message that you want to send with some other utility, or maybe as a file. Then you send it to the standard in of the wall command. In the example you see here, we use the echo command to send this text stream to the input of the wall command: "The system is going down in five minutes." When I hit enter, the message is displayed at the terminal of all logged in users.

Summary 7:22-7:34

That's it for this lesson.

In this lesson, we talked about the options for shutting down or rebooting the Linux system from the shell prompt.

We talked about how to do this by changing runlevels or boot targets.

Then we reviewed the halt and reboot commands.

Then we ended this lesson by looking at the shutdown and the wall commands.