4.4.2 Manage System Services

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

Manage System Services 0:00-0:49

In this demonstration, we're going to talk about managing services running on a Linux system. These services are also called daemons. They run in the background and they provide functionality either to the local running system or to another host somewhere on the network.

In earlier versions of Linux, we used the init daemon and init scripts to start, stop, restart, enable, or disable a Linux service. That's no longer the case. In the last couple of years, almost all distributions have switched to the systemd daemon, and so init scripts are no longer used.

Instead, we use the systemctl command. Let's look at how we do that. First I need to switch to my root user account, because only root can manage system services.

NTP daemon 0:50-3:22

On this system, I've install the NTP daemon. The NTP daemon is used to synchronize the time on my local system here with other computers on the network. That way all of my computers on my network have exactly the same system time. That's very useful for logging, for timestamps, and so on.

Well, one of the first things I need to know is whether or not that service is running, because if it isn't running, it's not synchronizing time with the other hosts on the network. Let's use the 'systemctl' command.

Then we'll use the 'status' option to see whether or not the NTP daemon is currently running or not. Press Enter, and it tells us that it is current inactive. It's dead. That means it's installed on the system, but it isn't doing anything, so I need to start it.

To do that, I use the same 'systemctl' command, but instead of using the status option, I use the 'start' option. This command tells the NTP daemon to go ahead and load up and start up on the system.

Now if I do the 'status' command again, we see that it is active and running. The daemon is now running on the network and doing its job, which in this case is synchronizing time with other network hosts.

There may be times when you have a particular daemon running on your system, and you decide, "Well, you know what? I really don't want it running anymore, I need to shut it down." In which case, you use the 'systemctl' command again, but instead of start we enter 'stop'.

This tells the NTP daemon to go ahead and shut down. If we do a 'status' again, it's no longer running. It's inactive. Let's go ahead and 'start' it again. Now the NTP daemon is running.

There may be times when you are managing a particular service and you go into its configuration file, and you make changes to that configuration file. The configuration file for most services is read only when that service initially starts.

If the service is already running and you make changes in the configuration file, they won't actually be applied until the next time you boot the system when that service starts. That probably won't work, because more than likely if you're in there making changes to the config file, you want those changes applied right now, and these cases you'd need to restart the daemon.

There's two ways to do this. One would be to go ahead and do a systemctl stop, and then do a systemctl start. That would reload the configuration. Another way to do it is to use the 'restart' option with the 'systemctl' command.

This will stop the NTP daemon, and then start it up again. Hit Enter. It will take just a second. Now if we do a 'status', we should see that it is running.

Reload Option 3:23-6:47

There's a third option for reloading the configuration, but it's not supported by all network services. You can also use the 'reload' option. With the reload option, we don't actually shut down the service and then bring it back online. Instead, we leave the service running and simply give it a signal to go and read its configuration file and make the necessary changes.

As I said, not all services support this option. In fact, I'm not sure if NTPD does or not; let's find out. And it doesn't like it. Not all services support the reload option, but some of them do. All of them will respond to the restart option, however.

Up to this point, we've talked about how to start a service. We've talked about how to stop a service. We've talked about how to restart a service and to view its current status. You can also use the systemctl command to either enable or disable a particular service on your system.

You're asking, "What's the difference between starting and stopping a service, and enabling and disabling a service?" The key is whether or not that service will start when the system is booted.

In fact, if we look right up here in the output of our status command, notice that it states, disabled. It's currently running on the system. It's not disabled in the sense that it's not functioning--it is functioning.

It's providing its network service right now, synchronizing time with the other hosts on the network, but if I were to shut down this system and turn it back on, the NTP daemon would not be loaded, because it's disabled.

However, I can reconfigure it such that this daemon is started every single time the system is booted. That way, I don't have to go in and manually start it. If I leave it at a disabled state and I want it to run, then I would have to go in and manually start it.

If you want to just check and see whether a service is enabled or not on system start up, you can use the 'systemctl' command, and use the 'is-enabled' option, followed by the name of the service in question, 'ntpd', and as we saw up here in the output of the status command, it is currently disabled.

Let's go ahead and enable it so the NTPD does start every time the system starts. Very easy to do, just replace 'is-enabled' with 'enabled', and then the name of the service that you want to enable. In this case, 'ntpd'. Notice when we did that, something very important happened.

The output of the systemctl command tells us that a symbolic link from /etc/systemd/system/multi-user.target.wants/ntpd.service has been created that points to usr/lib/systemd/system ntpd.service, which is just a fancy way of saying whenever the system boots up, NTPD is going to be loaded by the systemd daemon. Now we can use the 'is-enabled' command again, and it is enabled.

If for some reason you have a service running on your system that is enabled--in other words, it's starting every time the system boots--but you don't want it to do that anymore, you want to disable it so that it does not start automatically on system boot, you can just replace 'enable' with 'disable', followed by the name of the service in question.

Hit Enter, and you can see here that the symbolic link has been removed. If we check the status, we see that it is now disabled.

Summary 6:48-7:03

That's it for this demonstration. In this demo we talked about how to manage services. We first looked at using the systemctl command to start, stop, restart, and check the status of system services. Then we talked about how to use the systemctl command to either enable or disable a service on system boot.
