3.1.2 System Design Part 2

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

System Design Part 2 0:00-0:41

In this lesson, we're going to discuss how you go about planning a Linux installation. Now, this process includes several key tasks, such as conducting a needs assessment, selecting a distribution, verifying system requirements and hardware compatibility, planning your file system, selecting software, identifying user accounts, gathering network information, and selecting an installations source.

In this lesson, we're going to review the last five steps of the system design process. The previous three steps were covered in a separate lesson. Let's begin by looking at step four of the system design plan, where you specify which Linux file system is going to be used.

Selecting a File System 0:42-1:55

When planning a Linux implementation, you need to include specifications that will identify which file system is going to be used on your system's hard disk drives. With Linux, you can customize how your disk is going to be partitioned, and you can also define which file system you want to use. When conducting disk I/O operations, the operating system needs to know where data is stored on the drive, how to access it, and where it's safe to write new information on the hard disk. This is the job of the file system. The file system reliably stores data on the hard drive and organizes it in such a way that it can be found easily.

Most Linux distributions allow you to select from a wide variety of file systems that you can choose. Some are better than others. Some have different advantages over others. You can pick from ext2, ext3, Reiser, ext4, and btrfs. Most distributions will allow you to choose which file system you want to use when you initially partition your system's hard disk drives during the installation process. So, you need to already know which file system you want to use before you get to that point. When you're planning your Linux implementation, you need to specify which of these file systems you're actually going to use in your deployment plan.

Disk Partitions 1:56-2:27

This circle that you see on the screen represents the total space available on your hard disk drive. You can define these partitions within that space. A partition is, really, just a logical division of your hard disk, such as something like this, where we have two partitions on the drive. This is partition one. This is partition two. A hard drive could be configured with just a single partition that encompasses the entire hard drive, or you could configure it with multiple partitions: three partitions, four partitions, and so on.

Default Disk Partitions 2:28-3:49

When you're implementing a Linux system, you must use at least two partitions. However, you could, and probably should, create many more. These partitions really need to be defined during the initial installation. This is because changing your disk partitioning after the system's installed is challenging and time-consuming. It can be done, but it isn't fun. Therefore, it's a best practice to actually plan out how you will partition your hard drive before you start the installation process.

By default, most Linux distributions will propose the two partitions shown here during the installation process.

The first is a swap partition. This partition is used for virtual memory by the Linux operating system. The appropriate size for this partition will depend upon how the system is being used. For a desktop system, the general rule of thumb is that the swap partition should be about two times the size of the system RAM. This is because desktop systems usually run a large number of end-user applications that can be easily swapped to disk. So, for example, if your desktop system had four gigs of RAM installed, then you should have a swap partition that is about eight gigs in size.

Server systems, believe it or not, typically don't need quite as much swap space. Usually, a swap partition that's about one time the size of the system RAM is sufficient.

[Section Title] 3:50-4:10

In addition to the swap partition, you also need the root partition. This partition is mounted at the root directory, forward slash (/), of the file system. All of your user data, all of your applications, all of your logs, all of your configuration files are, by default, contained in this single

disk partition.

Recommended Partitions 4:11-9:00

In planning your Linux partitions, you really should consider creating many partitions on your hard disk drives. You might be asking, "Why on Earth would I want to do this?" It's because using more partitions adds a degree of fault tolerance to your system. The issue here is that any problem encountered in one partition is isolated from the other partitions in the system. When you're planning partitions for your Linux implementation, you should consider creating partitions for the mount points in the file system shown here.

First of all, you will need a partition mounted at the root directory. For most distributions, they're going to say that should be between four and eight gigs in size. That is way too small. That's got to be a lot bigger. The issue here is that if the partition that's mounted at the root of the file system is too small, over time, all the system updates that are going to get installed will use up that space, and you're going to run into problems. So make this really big.

You also need a small partition for the boot directory. This usually contains your Linux system files. As such, it really doesn't require a whole lot of space. The usual guideline is that it be about 100 to 200 megabytes in size.

You should also create a separate partition for the /home directory because this is the directory where all of your users are going to save their files. You have to allocate a lot of space to this directory, enough to accommodate all of your users' data. And I bet you'll find that no matter how much space you allocate to this partition, it will never be quite enough.

You also should consider creating a separate partition for the opt directory. You also might want to

create one for the /temp directory. This one should be about one to two gigs in size. This directory is automatically cleaned out every so often by the Linux kernel, it doesn't need a whole lot of space.

You should also create a separate partition for the /usr directory. This is where all of your system utilities are stored. The minimum they usually recommend is about five to ten gigs. That's, again, way too small. Make this a lot bigger so there's room for additional packages to be installed on the system. If you make it too small, you will probably find that, very quickly, you run out of space and aren't able to install new software on the system.

This one is really is important. If you don't do anything else, make separate directories for home and var. Everything else--boot, op, temp, and usr--are kind of optional. But I always create separate partitions for home and var. The issue here is that in /home, users will download tons of very large music and video files and consume up all the space. And /var tends to consume a lot of space, too, because this is where your log files get stored, and they can become quite large. If you have dedicated partitions for these two mount points, you actually protect your system from crashing.

Because if users download too many files or if your log files get too big, it is possible for those files to consume all of the available space on the hard disk drive. If they're on the same partition as your root directory or the /usr directory, then it is possible that your system will stop running. By isolating them to their own partitions, you keep that from happening.

Once you've planned out your partitions, the next task you need to complete is to specify which software you want to install on your Linux system. Understand that your Linux distribution will probably include a fairly extensive sampling of packages during the installation process. And because so many packages are available with most distributions, you might be tempted to install lots of software on that system, because it's free. But you really should not do this. Installing excess software on your system could, potentially, create security issues.

A much better approach is to use your deployment plan to identify the role that that system is going to play, and then, using that information, identify just those packages that are necessary for it to fulfill that role.

The general rule of thumb is, install only what is needed to do the job, and nothing more.

Then, once this part of your plan is complete, you need to specify which user accounts need to be created on the system. Remember, Linux is a true multi-user operating system, and that means one single system can include many different user accounts. Multiple users can actually use the same system at the same time over a network connection. Therefore, when you're planning to install your Linux system, you need to identify which user accounts need to be created on the system, and the installation utilities used by most Linux distributions will provide you with the ability to create these account during the installation process, although you could create them afterwards anyway.

Also, remember that root is one user that you do have to plan for. It will be created automatically for you during the installation process, but you need to remember that it's there. And you also need to make sure that you protect the password that you assign to the root user very carefully.

IP Address Assignments 9:01-10:18

Now, once this information is in your plan, you're ready to move on to the next step, and that is gathering network information.

In today's network world, most of the Linux systems you're going to be working with are probably going to be connected to some type of network. Therefore, you need to gather whatever network information is necessary in order for your system to connect to that network.

For example, you need to decide whether or not you're going to use a static IP address or if you're going to use an address that is dynamically assigned by a DHCP server.

Generally speaking, you're going to see workstations assigned an address using DHCP because we don't care if the IP address changes all the time, nor do we really go around to every single workstation in the organization and manually assign IP addresses. It takes a lot of time, and it's a management nightmare.

Servers, on the other hand, probably will have a static IP address assigned. That's because with a server, it's usually not a good thing if the IP address is constantly changing day to day, as could potentially happen with a DHCP configuration. So, for servers, we typically assign a limited number of static IP addresses. When you do this, you'll need to determine the appropriate IP address, the appropriate subnet mask, the appropriate default gateway router address, and the correct DNS server address.

Other Networking Parameters 10:19-11:10

There are a couple of other parameters that you need to specify in your plan, such as the hostname to be assigned to the system. Each host on the network needs to have a unique hostname.

We also need to specify which DNS domain is the system going to reside within, such as CorpNet.com.

We also need to make sure that a host firewall is configured. This is critical because it prevents other systems on the network from establishing an unwanted connection to your system, which is typically only done by somebody who is trying to exploit that system.

However, be aware that if that system is going to be hosting services that will be accessed by other hosts on the network, then you need to create a list of what we call exceptions. These are open ports in the firewall. For example, if it's going to be a web server, you have to open port 80. If it's going to run SSH, then you need to open port 22, and so on.

Installation Options 11:11-12:36

With your network configuration in place, you're ready to move on to the last part of the deployment plan, and that is selecting where you're going to install Linux from. There are many different options. Most Linux distributions will allow you to install locally from an optical disc. This is a very common way of deploying Linux. It's also kind of a slow way to deploy Linux, especially if you're deploying a lot of systems all at once.

If you're going to be installing a lot of systems all at once, a better option is to install remotely from a network server. With some distributions, you can install directly off the Internet from a package repository. If necessary, you can complete a remote installation using a VNC protocol.

For your deployment plan, you need to determine which installation source it is you're going to use and prepare the prerequisite hardware if necessary. Generally speaking, if you're going to be installing one or two systems, this is a great option. If you're going to be installing five or ten systems at the same time, you might want to look at the second option.

Once you've completed this step, your Linux deployment plan is now complete, and you should now have all the information you need to complete the installation in an organized and efficient manner. Before we end, please remember to file your deployment plan in a safe place, where you can find it again, because this information could be an invaluable step for yourself or other system administrators who may need to work on these systems you just deployed at some point in the future. They might need to know why you decided to configure them the way you did.

Summary 12:37-12:46

That's it for this lesson. In this lesson, we reviewed the last five steps of the system design process. We talked about planning the file system, selecting software packages, identifying user accounts, gathering network information, and then selecting an installation source.
