8.4.3 Swap Area Management

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

Swap Area Management 0:00-4:43

If you want to create a standard Linux file system to store data in the form of files and directories, then you use the mkfs command to create a file system on the given partition.

However, be aware that you can also format a partition such that it can be used for virtual memory on your Linux system. In this case, you need to format the partition as a swap partition, and this is done using the mkswap command at the shell prompt.

Understand that your Linux system can use virtual memory to allow it to run more programs than it has physical memory to support. For example, let's suppose we have this Linux system here and this green box represents the total amount of RAM installed in the system.

The operating system itself is going to use a big chunk of that available RAM just to provide operating system functions and then each application running on the system is going to use a chunk of memory as well. Now this is just a basic representation. There are many, many, many more processes running on the system besides the four that we've drawn here.

In this situation, you can see that most of our physical memory has been consumed. We have just a little tiny bit at the top there that can still be used for programs. Let's suppose I need to run another program, and this is a fairly big program. It's going to consume that much memory, so if I try to run this process in the memory available, it won't work, it's too big.

In order to make this work, what Linux does, and most other operating systems for that example, is take a look at the various processes that are already running in memory, and the kernel will decides which of those processes isn't really doing a whole lot at the moment, because if you think about it, there may be hundreds of processes running on your Linux system of which, a lot of them are just sitting there waiting for something to happen.

In which case, they don't actually need to consume our physical RAM, so what we do is take those processes, and we move them out of physical RAM and put them onto a hard disk drive. Essentially, we map memory locations that don't exist in physical memory, but instead are over here on one of our hard disk drives in a partition on the hard disk drive.

Now the processes themselves that are being swapped don't realize that they are not running in memory anymore, they just think they're being moved to a different memory location. The process itself has no idea that that memory location is actually in a hard disk drive.

The hard disk drive is much slower than physical memory, and so we don't want to put anything in the swap area on the hard disk drive that's being actively used, because it's going to run painfully slow but for a process that's just sitting there waiting for something to happen, that works just fine.

By doing this, we free up space, and so the additional application that we want to run is able to be inserted into physical memory and run. Then if this red application needs to be used again for some reason, maybe a certain event has happened so that's triggered it to start doing things again.

Then we will take it back out of the swap area and put it back into physical memory and we'll look for some other application that currently isn't doing anything.

Maybe that application we just loaded is done with its job and it's just sitting there waiting for something to happen so we can then take it out and put it into the swap area. That's why we call this swap, because we're constantly moving processes in and out of that area. By doing this, the system can run more processes than it physically has RAM to support.

On a Linux system, you can use either a partition or an entire hard disk for this virtual memory, for a swap area. Although my experience has been that you rarely use an entire hard disk, that's just way too much space.

Instead what we do is define a partition on the hard drive and use it for virtual memory. Be aware that a Linux system can actually use more than one swap partition. In fact, if you have a very heavily used system, having multiple swap partitions can be really, really beneficial if you put those swap partitions on multiple storage devices.

In this case, I'm putting an extra hard disk in the system. I'm defining a swap area on it as well as on the main hard disk and by doing this, the system has two places where it can swap data back and forth from physical memory to the swap partition.

By doing this, we can actually speed up the system a fairly considerable amount depending upon how heavily it's used. If you have a lightly used system, using multiple swap areas isn't all that effective but on a heavily used system like a server, having two or three or more swap areas can really pick up performance.

mkswap command 4:42-5:44

Now, just like a data partition, a partition used for virtual memory for swapping has to be prepared with the mkswap command before you can actually use it. Of course, before you do this, you have to create the partition first and you also need to set the partition type to type 82. The default if you use fdisk or gdisk is to create a type 83 partition which is a Linux data partition.

You use it for swap memory, you have to first use the fdisk utility or the gdisk utility to change its type to 82 for fdisk or 8200 for gdisk. As you can see, that changes its file system type to a swap partition.

The syntax for using mkswap is shown here. We enter mkswap followed by the device file name for the partition that we want to format as a swap partition. For example, if you had a logical partition, sdb6 that you wanted to format as a swap partition, you would enter mkswap and then the device file name /dev/sdb6.

---

swapon 5:43-6:36

After you create the swap partition with the mkswap command, you then need to activate it. This is done using swapon. You enter swapon followed by the device file name of the swap partition you just created with mkswap.

In this case, we're entering swapon /dev/sdb6. If you want to view a listing of all the swap areas that have been defined and enabled on your system, use the swapon command with the -s parameter, then you can see the swap areas that are currently being used on your system. In this case you can see that I have two swap partitions on two different storage devices.

You can also see the priority of those devices over here. The one with the priority of -1 is going to be used primarily and the one a value of -2 will be used as the secondary one. Basically, it will be used only if for some reason this first one is busy and can't be used.

---

swapoff 6:34-7:24

After you've created and enabled a swap area, for some reason you need to disable it, you use the swapoff command. As you can see here, swapoff followed by the device file name of the swap partition.

Also be aware that if you have multiple swap partitions on your system and you want to activate them all at the same time, instead of going through one at a time, then you can just use the swapon -a command and that will just go through your /etc/fstab file, find all the swap partitions that are contained therein and immediately enable them and put them into service.

You can do the opposite if for some reason you need to disable all of your swap partitions which frankly, I don't think I've ever done. You can use the swapoff command with the -a and that will again go through fstab file and deactivate all of your swap partitions.

Summary 7:24-7:32

That's it for this lesson. In this lesson we reviewed how you manage swap areas on a Linux system. We first looked at the mkswap command, then we reviewed the swapon and the swapoff commands.
