12.4.1 Routing Configuration

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

Routing Configuration 0:00-0:08

In this lesson, we're going to review how to configure routing information on a Linux system.

How Routing Functions 0:09-5:39

Let's begin by actually discussing how routing works on an IP network. Understand that routers operate at the network layer of the OSI model and are used to connect various networks together.

Then, using the IP protocol, routers do just what their name implies; they route data across multiple networks and deliver information to a destination host. In this example, we have a host over here, and it needs to deliver data to a host over here.

These hosts are connected together by multiple routers that route IP traffic. To get the data from host A over here to host B over here, the data is going to be passed first to this router, and then this router will decide where to send it next, and then this router will decide where to send it next, and this router will decide where to send it next, and then actually deliver it to the destination system.

The router hardware itself-- for example, this one-- could be just a computer system that has two or three network interfaces installed, one connected to each network segment. Or, more than likely, it's going to be a specialized hardware appliance that does nothing but routing.

One of the key jobs performed by these routers is to determine which is the best way to get information to the right destination host.

For example, we have host A again here and host B. We need to get data from A to B, but notice that there are multiple routes that the data could take. It could go to this router to this router to this router to this router, and then to the destination.

Alternatively, it could go from this router to this router to this router to this router to this router. Another option would be to send it to this router and then to this router and then to this router and then to this router and then to the destination.

There's even a fourth option. We could send it to this router and then to this router and then to this one, then back down to this one and then to this one and to this one and to this one. Which one of these routes is the best?

Well, there are actually a lot of variables that come into play to determine which route is the best. And those variables can change minute to minute, meaning that one route might be the best at one time, and a different route might be best later on.

In order to do this, each of these routers maintains a routing table. This routing table lists all of the available routes. Routers use various algorithms that evaluate such things as distance, cost, network bandwidth, network latency, network status, and so on, to determine which of these routes is the best way to get the data to the destination host.

Here's an important thing that you need to understand. Every Linux system, regardless of whether it's a workstation or a server or whatever, also maintains a routing table within its memory that determines where to send data on a network. It does not use routing protocols to determine the best route; that's the job of the router.

But it does have a listing within its router table that says where it needs to send what kind of data. When you're configuring networking parameters on a Linux system, one of the key tasks you need to perform is to configure the default router address.

The default router is the default location where packets are sent to if they're addressed to a host that doesn't reside on the local network segment. Here we have a network over here, that's the 192.168.1.0 network, and over here we have a network that is the 10.0.0.0 network.

Let's suppose that this host right here, host A, needs to send data over here to host C. Well, when it creates the IP packets, it sets as the destination address on those packets the IP address of this host over here 10.0.0.2.

Well, when it goes to send that packet out on the network, it takes a look at the destination address, specifically at the destination network address. And it says, "Hey, you know what? That destination host resides on the network of 10.0.0.0. I'm on 192.168.1.0. Um, that host isn't over here."

It knows because the network address is different, that the host resides on some other network. It doesn't know what it is. In order to get it there, this host is going to send that packet to the default gateway router. How does it know what the default gateway router address is-- in this case, 192.168.254? It's configured on the system.

You have a default router gateway address parameter and you put it in. You say, "Okay, if you don't know what to do with a packet because it resides on a different network segment, send it to 192.168.1.254." On the other hand, if A needs to send a packet down here to B, which does reside on the same network as A, then the default gateway is not used.

When 192.168.1.1 creates the IP packets that are going to go to 192.168.1.2, it takes a look at them and says, "Hey, I know where that guy lives. He's on my same street. He's on my same network--192.168.1.0. I'll just send it straight to him." And it does not send it to the default gateway for routing.

Role of etc/sysconfig/network/route File 5:40-9:08

Your default gateway router address is stored in the /etc/sysconfig/network/routes file on most distributions. Other distributions might save it in a different place. The syntax for specifying the default route is shown here.

We enter default to indicate that it is the default gateway route, and then we specify the IP address that we want those packets sent to if they're addressed to a system that is on a different network other than the local network.

In this case, we're setting the default gateway to the router with an IP address of 10.0.0.254. And we can also specify the network adapter that this route applies to, because it is possible for a Linux system to have two, or maybe even three, different network interfaces installed, and they could potentially all be connected to different networks.

Therefore, they have different gateway routers. As such, we would have to put individual entries within this file for each network interface.

You can define additional routing information in the /etc/sysconfig/network/routes file. In the previous file we just looked at, you can specify only the default gateway. In this file, we can put routes to all kinds of different networks beyond just those that go to the default gateway. The syntax is shown here.

We enter the destination, then we specify which router the packets being addressed to this network should go to. In this case, any packet addressed to a host on the 192.168.2.0 network needs to be sent to the router that has an IP address of 10.0.0.1.

The idea here is that this router knows how to get whatever data we're sending to it over to this network and to the destination host. The netmask parameter specifies the netmask for the destination network over here. In this case, it is 255.255.255.0.

Then the fourth column specifies which interface we're talking about. In this case, we're applying this route to the ens32 interface. If you don't specify an interface, then the route will be applied to all interfaces.

Before we go on, I do need to point out that you can also configure the default gateway in this file as well. What we would do is, instead of specifying a network address in the first column, you would instead specify the text default.

By specifying default, we're just saying any network that we don't know how to get to will go to this particular router. In this configuration, we're sending packets addressed only to the 192.168.2.0 network to this particular router.

Notice that there is a fifth column over here-- the TYPE column. It is an optional column, and it's used to specify the route type.

Here's an important point that you need to remember, and that is the fact that if you modify this file-- say, open it up in a text editor, add a new route to it-- it's not going to be applied until your network interface is restarted. In other words, this file is not read dynamically every time a routing decision needs to be made. It's read only when the interface comes up.

Therefore, if you make a change, you'll need to take your interface down, such as with the ifdown command, and then bring it back up with the ifup command. At which point, the routes file will be reread, and your routing table in memory will be updated with the new information.

---

route 9:09-12:23

You also need to be familiar with how to manage routes using the route command. You can use the route command to either display or to modify the routing table on your Linux system.

If you enter route without any options, such as we've done here, all it does is display the current routing table. Here, you can see we have an entry for a default route. Any packets that we don't know where to send to because we're not sure where the network is, we're just going to send it to the router with an IP address of 10.0.0.0. That applies for our ens32 interface.

You can also add routes to the host routing table using the route command. The syntax is shown here. You enter route add -net, followed by the network_address of the route that you want to create; netmask, followed by the subnet mask of this network over here; gw, which stands for gateway, followed by the name of the router that connects us to that network.

I'll show an example down here. We enter route add -net, and we're going to create a route to the 192.168.2.0 network, which has a subnet mask of 255.255.255.0. To get to that network, we need to send packets to the router, the gateway (gw) on our network segment that has an IP address of 10.0.0.1, and we'll let that router figure out how to get those packets over to this network.

After I'm done, I run the route command again, and you can see the new route that's added. Notice we have two possible entries here. We have the default route, and we also have a route specifically to this network.

If my system creates an IP packet and addresses to a host on the 192.168.5.0 network, it's not going to use this route. It is instead going to use this route, the default route, and it will send it to this gateway.

If, on the other hand, I create a packet--an IP packet-- and I address it to a host on the 192.168.2 network, it will be sent to this router instead, not to the default gateway.

You could also use the route command to remove a route from the routing table. To do this, you enter route del instead of route add, and the rest of the parameters are the same. In this example, I'm deleting the route to the 192.168.2.0 network, and I specify the same parameters that I did before for the netmask and the gateway. Then, when you run route, you'll notice that the route that we had before to that network is now gone.

You can also use the route command to set the default route. The command is slightly different than what we did before. We enter route add, and then we specify default, because we're not specifying any particular network segment.

Instead, we're saying all network segments that we don't know how to reach, so we just put default. Then we enter gw, followed by the address of your default gateway. An example is shown here: route add default gw 10.0.0.254.

Please be aware that any changes you make to your routing table with the route command are not persistent. If you want the route changes to be persistent across reboots, you really need to go in and edit your /etc/sysconfig/network/routes file and make them statically there.

ip 12:24-13:37

The last thing we're going to look at in this demonstration is how to configure routes with the ip command. For example, if you want to view your routing table, you would type ip route show at the shell prompt.

You could also use the ip command to add a static route to the routing table. In which case you'd type ip route add, followed by the network address and its prefix. Then you type via, and then you enter the IP address of the router that you want those packets sent to, dev, and then the name of the network interface.

An example of that is shown down here, where we enter ip route add 192.168.4.0/24, to specify the subnet mask (the prefix), via, and then the IP address of the router that I want those packets sent to to reach this network, which is 10.0.0.254, dev, followed by the name of the network interface this route applies to, ens32.

We can see the route added when we use the ip route show command. Here's the new route. You can also delete a route with the ip command. The syntax is ip route del, followed by the network address and its prefix that you want to remove. When you do, it's removed from the routing table.

Summary 13:38-13:50

That's it for this lesson. In this lesson, we discussed how to manage route settings on a Linux system. We looked at the role of the /etc/sysconfig/network/routes file. We talked about how to use the route command, and then we ended this lesson by talking about how to manage routing using the ip command.