

12.2.2 Address Configuration

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

Address Configuration 0:00-0:31

After you install a network interface in your Linux system, then you need to configure four different parameters on that interface in order for that system to participate on an IP-based computer network. You need to assign an IP address, you need to assign a subnet Mask, you need to define the default gateway router address, and you need to assign the DNS server address. In this lesson we're going to focus on just the first two parameters--assigning the IP address and assigning the subnet mask.

Address Assignment Options 0:29-2:54

There are actually two different ways to accomplish this. The first one is to use a static IP address assignment, and the second one is to use a dynamic IP address assignment. In a static IP address assignment, you manually configure the network host with an IP address and a subnet mask. If you do this, then the IP address stays static. It never changes. This is the preferred option if you're working with a network server, because we don't want server IP addresses changing.

However, we typically do not use static IP address assignment for workstations. This is because if you use a static IP address assignment, then that host is going to consume that IP address, whether the system is in use or not. It also requires a lot of legwork on the part of the system administrator. The administrator has to visit every single computer and manually specify the IP networking parameters. Generally speaking, you will not use a static IP address assignment for workstations, while you will use a static IP address assignment for servers.

The opposite is true for dynamic IP address assignment. Using a dynamic IP address assignment, the computer will actually go out on the network and contact a dynamic host configuration protocol server, or DHCP Server, when that system first boots up. Then the DHCP server will automatically assign whatever IP address is currently available to that host for a specific amount of time. That's called your lease time.

This option is almost always used for workstations. It makes it really easy to configure IP networking parameters on a very large number of network hosts without really having to do anything. You just turn the workstation on, and it gets its IP addressing information. And it also conserves IP address usage as well, because addresses that are used by systems that are powered off can be reassigned to another network host that currently needs it.

This option is not typically used for servers. This is because the address that gets assigned to a particular host can change, and it can change frequently. This is not a good thing for infrastructure systems like servers. They need to have the same IP address all the time. Works fine for workstations, just not servers.

Viewing Addressing Information with ifconfig 2:54-5:10

Let's talk about how you would go about configuring an IP address statically on a system such as a server. One way to do this is to use the ifconfig command at the shell prompt. Notice here that if you type 'ifconfig' without any options, all it does is display the current status of all the network interfaces in the system. You'll notice in the output that I actually have two network interfaces listed. I have ens32 and I have lo. ens32 is the Ethernet interface that is installed on my system that's connected to my network switch.

The lo interface, on the other hand, is not a real network interface. There is no physical interface in the system assigned the alias of lo because lo is a virtual network interface. It's the local loopback adapter. Notice that it's assigned a special IP address of 127.0.0.1. It's only used for internal communications between network services running on the system. It's not actually used to communicate on the network.

Be aware that this is required for most Linux services to run properly, so don't do anything dumb like delete it. It's got to be there. You also don't need to worry too much about its configuration. You can focus up here on the one that's actually connected to the physical network.

Notice in the output of ifconfig that several important network parameters are listed. First of all, we have the HWaddr parameter, which is the MAC address that's been assigned to the network board. We have the inetaddr, which is the IP address that's been assigned to the network interface. We also have the Bcast parameter, which specifies the broadcast address of the network segment. We have the Mask parameter, which specifies the subnet mask that's been assigned to the interface.

Down here we have RX packets, which specifies statistics for the received packets on the interface. We also have TX packets, which contains statistics for transmitted packets from the interface. Collisions down here tracks the number of collisions detected by the interface, which on a modern Ethernet network should be zero--it is. We also have RX bytes, which tells us how many bytes have been received by the interface, and TX bytes, which tells us how many bytes have been transmitted by the interface.

Configuring an Address with ifconfig 5:10-5:55

In addition to displaying information about a network interface with `ifconfig`, you can also use it to assign IP networking parameters. The syntax for doing this is shown here. You type `ifconfig`, followed by the interface you want to make the assignment to, and then the IP address that you want to assign. Then `netmask`, followed by the subnet mask you want to assign. Then `broadcast`, followed by the broadcast address of the network segment. An example is shown down here. We're assigning the `ens32` interface an IP address of `192.168.1.1`, a subnet mask of `255.255.255.0`, and a broadcast address of `192.168.1.255`.

Interface Configuration File 5:51-8:35

It's important to note that if you make a static IP address assignment with the `ifconfig` command, that assignment is not persistent. If you were to remove the system, it would be lost. In order to make it persistent, you need to configure a special file in the `/etc/sysconfig/network` directory. In this example, the file that we're going to use to configure a static IP address on an Ethernet interface is named `ifcfg-ens32`. Be aware that this is the way it works on the openSUSE distribution. On other distributions, this file might be named differently. For example, it might be named using the MAC address of the NIC. You'll just have to check and see how your particular distribution names the interface configuration file.

This file is used to configure IP addressing parameters on that interface when the system turns on. The first parameter you need to be familiar with in this file is `BOOTPROTO`. Right now it's set to `dhcp`, which specifies that we dynamically assigned this interface and IP address from a DHCP server. You could change this to a value of `static`, which would specify that the interface use a static IP address assignment. We also have a parameter down here called `STARTMODE` and it's currently set to `auto`. This option specifies that the interface be brought online automatically when the system is booted. If you set it to a value of `manual`, then you would have to manually start the interface instead of automatically starting it.

If we have specified `BOOTPROTO=dhcp`, we really don't have to do a whole lot else with this configuration file, because that will be taken care of by the DHCP Server. On the other hand, if we specified a static IP address assignment in `BOOTPROTO`, then we would have to go through the rest of these options and configure the appropriate parameters. For example, `IPADDR`. We would have to specify the IP address that we would want to assign to this interface, along with the subnet mask under the `NETMASK` parameter. We would also have to specify the network address of the network segment. We would also have to specify the broadcast address of the network segment as well.

There are some additional parameters in here as well that you could modify if you need to--it's generally not necessary--such as the `MTU`, where you can set the size of the maximum transmission unit. This is always set to 1500 by default on Ethernet networks. There are situations where you might need to change this. For example, if your network uses jumbo frames, then you could set this parameter to something like 9000 instead of 1500. This is not done very often, though.

Restarting a Network Interface 8:33-9:04

If you make any changes to this file, you have to restart the network interface in order to apply the change. Otherwise, it won't be applied until the next time you boot the system. One simple way to do this is to use the `ifdown` command to take the interface down, and then bring it back up with the `ifup` command. An example of doing that is show here. We enter `ifdown ens32`, followed by `ifup ens32`. At which point, the new configuration parameters you specified will be applied to that interface.

Getting an Address Dynamically 9:03-10:12

If you want your network interface to go out on the network and get an IP address from a DHCP server, you can tell it to do that from the command prompt, using the `dhclient` command. If you've configured DHCP in your interface configuration file, then you can accomplish the same thing by using the `ifdown` and `ifup` commands. Because of your configuration in the interface configuration file, it will go out and talk to DHCP.

If you want to tell a running interface to go out and get an address from a DHCP Server, regardless of whether it's set to use a static address or dynamic address, you can just run the `dhclient` command shown here. The syntax is to enter `dhclient`, and then the name of the interface that needs to go out and contact a DHCP Server and get an IP address--in this case, `ens32`. Notice when we do that, that the interface first sends out a `DHCPREQUEST` packet and then it gets a `DHCPACK` back from the DHCP sever. Notice that the IP address of `10.0.0.97` was sent back from the DHCP server to the `ens32` interface to be its IP address.

Viewing Address Information with ip 10:09-10:45

Another command you can use to manage IP addressing on a Linux system is the `ip` command. In this example, I'm using the `ip` command to view my current networking configuration. I type '`ip addr show`', and it shows me my configuration for my loopback interface and also for my `ens32` interface. I can see many of the same statistics that I saw with the `ifconfig` command, such as my MTU Size, my IP address and Mask, my Broadcast address, and so on.

Managing IP Addressing with `ip` 10:43-11:23

You can also use the `ip` command to configure IP addressing parameters; to do that, you use the syntax shown here. You enter '`ip addr add`', and the IP address that we want to assign to the interface, '`dev`', and then the name of the interface. If we want to remove an IP address, we replace `add` with '`del`', and that will remove the IP address from the specified interface. If we want to disable an interface--basically, take it down--we enter '`ip link set interface down`'. Then to re-enable that interface, we do '`ip link set interface up`'. Basically, that's the equivalent of using the `ifdown` and `ifup` commands.

Summary 11:24-11:46

That's it for this lesson. In this lesson we talked about how to configure an Ethernet network interface in a Linux system. We talked about how you can use `ifconfig` to make a static IP address assignment. We talked about accomplishing the same thing using an interface configuration file. Then we looked at using dynamic addressing on an Ethernet interface, and then we ended this lesson by reviewing how you can use the `ip` command to manage IP addressing on a network interface.

Copyright © 2022 TestOut Corporation All rights reserved.