12.2.3 View and Manage IPv4 Addresses

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

View and Manage IPv4 Addresses 0:00-0:15

In this demonstration we're going to look at viewing and managing IP addresses from the shell prompt, and we're going to focus on two different command line tools in this demo. The first is the ifconfig command and the second is the ip command.

Use ifconfig 0:16-1:50

Let's look at ifconfig first. You can use the ifconfig command as a standard user, as I've done here. And if you do this, you'll be able to view your current networking configuration. However, if you want to use ifconfig to change your current networking configuration, then you do have to be logged in as your root user.

If you just type 'ifconfig' at the shell prompt, it doesn't make any changes to your current networking configuration, it just displays it. Really it's analogous in function to the ipconfig command on a Windows system, although there are some fairly significant differences, as you can see in the output here.

Notice that when I enter 'ifconfig' on this system, it shows I have two interfaces installed in this system. We have ens32 and lo. A lot of folks get confused when they initially run ifconfig and see two network interfaces. Because they think, "I've got only one network interface installed in this system. Why am I seeing two?"

Well, understand that this one up here is your standard Ethernet interface, either wired or wireless adapter. This one down here is a virtual interface, and it's the loopback adapter. As I said, it's a virtual adapter; it's not a hardware adapter. It's simply used by local services running on the system to send messages back and forth and it has to be present.

If you don't have a loopback adapter running, then a lot of services on your system won't run correctly. In fact, if you have a Linux system that doesn't even have an Ethernet adapter installed and you type 'ifconfig', you're still going to see a loopback adapter. It's installed by default on all Linux distributions.

View Interface Statistics 1:51-3:26

For our purposes today, however, we don't care that much about the loopback adapter. We want to focus on the Ethernet adapter that's been installed in this system. Notice that when I run ifconfig that many different statistics about that interface are displayed. Right here it tells us that it's an Ethernet adapter. You could have a serial adapter installed in the system too, in which case you would see Serial instead of Ethernet here.

HWaddr specifies the hardware address, the MAC address, of the network interface. inet addr, as you might guess, displays the IP address assigned to the system. Bcast identifies the broadcast address of the network segment that this Ethernet interface is connected to. Mask specifies the subnet mask of the network interface. If you have IPv6 enabled on the system, then you will see the inet6 addr parameter, which specifies the IPv6 address that's been assigned to this system.

You will also see the Scope of the IPv6 address. In this case, it's a link-local address. MTU specifies the MTU size. RX packets lists the number of packets that have been received by the system. Notice we also see the number of errors, drops, overruns, and frame errors. TX packets displays the number of packets that have been transmitted by this interface and also displays any errors, drops, and overruns that have been encountered with transmitted packets.

Also notice that there is a collisions parameter down here that identifies the number of collisions that have been detected on the network segment. We also see the number of bytes that have been received by the interface and the number of bytes that have been transmitted by the interface as well.

Configure IP Address Parameters 3:27-4:49

ifconfig works great for viewing information about your network interface, but it can also be used to configure IP addressing parameters. The syntax for doing this is to type 'ifconfig', just like we did before. But then we have to specify the name of the network interface that we want to manipulate, because it is possible for you to have two or three or maybe even four network interfaces installed in your system.

In this case, we need to specify the 'ens32' interface, and then we have to specify what we want to do. In this case, let's set the IP address to '10.0.0.160'. Because we're setting the IP address to a different address, we need to also configure all the other parameters that go along with it. First we have to specify the subnet mask that we used with this IP address, using the 'netmask' parameter. This network segment uses a 24-bit mask, '255.255.255.0'.

Next we have to specify what the broadcast address is for this network segment. We type 'broadcast' followed by the broadcast address, which in this case is '10.0.0.255'. Press Enter, and now let's run 'ifconfig' again without any parameters to view our configuration and we should see that the IP address has changed to 10.0.0.160. The broadcast and the subnet mask address didn't change at all because we configured this IP address on the same network as the old IP address that it had.

Use if Commands 4:50-7:00

There are two other related if commands that you need to be aware of. One is ifdown. The other is ifup. ifdown, as you might guess, disables the network interface. In this case, let's disable the ens32 adapter. I type 'ifdown ens32', press Enter, wait just a second, and the network interface is now down. If I type 'ifconfig' at this point, notice that we have only one adapter now, the loopback adapter; ens32 is not even listed. If I want to bring the interface back up, I can type 'ifup' followed by the name of the interface, 'ens32'. When I do this, something interesting is going to happen.

The manual IP address that I made earlier up here for 10.0.0.160 is going to be gone. Because when you run the ifup command, it's going to go read the configuration file for the interface that you specified, and it's going to assign the various default networking address parameters that are specified within the interface's configuration file. That file could contain a static IP address assignment, or maybe it's configuring the interface to use DHCP to go out on the network and get an IP address from a DHCP server.

To verify this, let's run the 'ifconfig' command again, and notice that our IP address has changed from 160 to 97. That's because this interface is configured by default to use a DHCP server for IP addressing, so when we brought it back up with ifup, it went out on the network segment, contacted the DHCP server, and got an IP address lease from it.

You can also manage IP addressing from the shell prompt using the ip command. For example, if we wanted to view our current networking configuration, we would enter 'ip addr show' to display our current configuration. It's the equivalent of running ifconfig. Here you can see our two interfaces, lo and ens32, and here's the IP address that's been assigned to the ens32 interface. Down here is also the IPv6 address that's been assigned to that same interface.

Use ip Command 7:01-7:53

If you wanted to, you could use the ip command to actually add additional IP addresses to your interface. To do this, you enter 'ip addr add', instead of show like we did before, and then you have to specify the IP address that you want to add. In this case, it would be '10.0.0.84', and we have to specify the mask that we want to use.

We're going to use cidr notation this time; we're using a 24-bit subnet mask. That's '255.255.255.0/24 dev', which stands for device, and then we have to specify which device or interface we want to assign the IP address to: 'ens32'. If we run 'ip addr show' again, we see that we have the 10.0.0.97 address that we had before, but now we have a second IP address assigned to the same interface--10.0.0.84.

Remove an ip Address 7:54-8:25

Just as you can use the ip command to add an IP address to an interface, you can also use it to remove an IP address from the interface. The syntax is almost the same as what we used before. We enter 'ip add', but then we replace add with 'del'.

Then we specify the IP address that we want to remove, and its mask 'dev' and the name of the interface that we want to remove it from. If we do an 'ip addr show' again, we see that we're back to the configuration we had before with just one single IPv4 address assigned, along with the IPv6 down here.

Change the ip Address State 8:26-9:15

Just as you can bring an interface up and down with ifup and ifdown, you can do the same thing with ip commands as well. For example, if we wanted to disable an interface, we enter 'ip link set', followed by the name of the interface, 'ens32', and then what we want to do--we want to set it to a 'down' state. At this point, the interface is down. Do an 'ip addr show'; we still see the interface listed, but notice over here it's in a down state. It's not listening, it's not responding to network requests, and it's not transmitting frames on the network.

To bring it back online, we use the same command again, but change it from 'down' to 'up', as you might guess. If we run the 'ip addr show' command again, we see that its state has changed to up and now it can process frames.

Summary 9:16-9:26

That's it for this demonstration. In this demonstration we looked at commands you can use from the shell prompt to manage IP addressing on your system. We first looked at the ifconfig command, we then looked at ifup and ifdown, and then we ended this demonstration by looking at the ip command.