**CentOS Networking**

Networking

A system administrator will need to know about the networking environment for their CentOS Server. Some of the commands will work on both Ubuntu and CentOS servers, and some of these networking commands are unique to CentOS. This server is networked on my home router with DHCP, which means the server automatically received an IP Address, subnet, and gateway. I think it is better to set the IP address to static so that it never changes. I prefer to keep the IP Address the same on a server as it is more consistent and easier when it comes time for troubleshooting issues.

**Default Settings**

A screen shot of a computer

Description automatically generated

The default network configuration on this server is DHCP, which means the server will automatically receive an IP Address from the modem as shown above. The command **ifconfig** shows network information such as IP Address, subnet, broadcast address, and MAC Address of each interface. The command **nmcli device status** shows the network interfaces and their respective status. On CentOS, these settings are loaded on bootup from a configuration text file located in this path **/etc/sysconfig/network-scripts/ifcfg-enp0s3.** Note that the end of that command is the only device / interface that is connected in the screenshot above.

A screenshot of a computer

Description automatically generated

Next, use the command **cat /etc/sysconfig/network-scripts/ifcfg-enp0s3** to view the startup network configuration settings. The line **BOOTPROTO=’dhcp’** is how the server is told to automatically obtain an IP Address upon startup. If we wanted to change this to a static address, we could change the settings with a text editing tool and change *dhcp* to *none* and then add the lines for the IP Address manually. We also have the option to change things like the proxy settings, the interface name, and whether to use these settings on boot.

A black screen with white text

Description automatically generated

Using the command **ip route** we can see the default ip routing settings for the connected interface. It is vital to confirm these settings are correct as the gateway is what determines the success of all outgoing traffic from the server. The default setting for this is setup through DHCP, but it can also be set manually within the text file referenced previously (**/etc/sysconfig/network-scripts/ifcfg-enp0s3**).

1. **Command – “sudo yum history”** **& “sudo yum history info 5”**

A screenshot of a computer

Description automatically generated

* The command shows the list of programs installs or updates to the server using yum, and the date the change occurred. To view more information about a specific installation, use the ID in the left column and add it to the end of the command. For this example, I will look for more information on ID 5 (see below).

A screenshot of a computer

Description automatically generated

1. **Command – “w”** **&** **“who”**

A screen shot of a computer

Description automatically generated

* A simple command to display the online users in the system. Alternatively, you can run the command **“who”** to show who is logged in, but it will show less information.

1. **Command – “ifconfig”**

A screenshot of a computer

Description automatically generated

* Short for “Interface Configuration”, this command shows which network interfaces are enabled, the IP, subnet, and MAC Address of each interface.

1. **Command – “cat /etc/resolv.conf”**

A screen shot of a computer

Description automatically generated

* This shows the current DNS settings for the network, this command is the same as on the Ubuntu server.

1. **Command – “sudo cat /etc/ssh/sshd\_config”**

A screenshot of a computer program

Description automatically generated

* To show the SSH settings for the CentOS server, run **cat /etc/ssh\sshd\_config.** It is important for system admins to confirm these settings are correct, as ssh being enabled always presents a security risk. Verify the settings are correct to ensure secure connectivity.

1. **Command – “netstat -tuln”**

A screenshot of a computer

Description automatically generated

* The **netstat** command is part of the **net-tools** package we installed previously. It can be used to check the open ports and ‘listening’ services on the server. The system admin should regularly check the open ports to find potential security vulnerabilities on the server. Make sure only necessary ports are open to keep out malicious activity on the server. The **-tuln** option with **netstat** shows TCP ports (**t**), UDP ports (**u**), listening sockets (**l**), and **n** will display the IP Address rather than the host name. It is good to note that some companies have very strict security compliance policies regarding which ports can be open. Be sure to document which ports are open and why. For instance, some apps and services may need to have specific ports opened to function correctly.

1. **Command – “sudo firewall-cmd –list-all”**

A screenshot of a computer

Description automatically generated

* Knowing the status of the Firewall is another thing admins will need to check and configure to ensure secure connections. This command shows the active firewall rules on the server and which ports and services are allowed.

1. **Command – “systemctl list-units –type=service –state=running”**

A screenshot of a computer screen

Description automatically generated

* Show all active network services with this command. Any unnecessary services should be disabled for better security and server performance. Some services could be outdated or redundant. To disable a service, run “**sudo systemctl stop *servicename”,* and then run “sudo sytemctl disable *servicename”.*** In the event of an audit, it is important to know which services are running and why to meet compliance requirements.

1. **Script:**

A screenshot of a computer program

Description automatically generated

The purpose of this script is to prepare a report containing vital information regarding the state of the server network settings. For example, this report could be used to audit the server once per week to make sure that it is compliant with company standards, and to ensure that the server is secure with no unwanted connections or running services. After running the script, use **ls** to confirm the output of the script is there. And use **cat *filename*** to view the full report.

A screen shot of a computer

Description automatically generated

The purpose of this script is to prepare a report containing every networking command I covered. This type of report can be used to audit the server and confirm the settings are consistent, and no undocumented changes have been made. After running the script, use **ls** to confirm the output of the script is there. And use **cat *filename*** to view the report.

First, I set a variable called ‘date’ which just stores today’s date in a variable, I did this so I can include it in the name of the report. I like the idea of users opening this report and knowing how current this data is. Without current and reliable data, it is very difficult to troubleshoot issues and keep up with server maintenance. I then created one more variable named “file” which stores the name of the file I am sending this report to. This made it more efficient for me so I could avoid typing out the long file name after each command.

With these commands, my goal was to cover the essential network settings needed for troubleshooting issues. If I run into a networking issue at work, the first thing I check is the network configuration of the client system that is trying to talk to the server. It is important to confirm that the IP Address, subnet, and gateway are set correctly. This covers the Network Layer (layer 3) of the OSI Model we learned in Networks class.

Next, it is vital that we confirm the state of the firewall and open ports. The system admin should regularly check the open ports to find potential security vulnerabilities on the server. If there are any unnecessary firewall rules in place that could present a security risk to the system with unwanted access to the server. Getting consistent updates and reports on both the firewall and ports is very important for keeping the server secure.

**Sources**

* <https://www.aholdengouveia.name/LinuxAdmin/networking.html>
* <https://phoenixnap.com/kb/configure-centos-network-settings>
* <https://www.reliablesite.net/hosting-news/centos-networking-basics/>