Linux Admin

Networking Lab

By Nicholas Mueller

Sources:

“How to Install and Use dig and nslookup Commands in Linux”

<https://www.tecmint.com/install-dig-and-nslookup-in-linux/>

“An Introduction to the ss Command”

<https://www.linux.com/topic/networking/introduction-ss-command/#:~:text=%20An%20Introduction%20to%20the%20ss%20Command%20,to%20ss%20is%20to%20have%20it...%20More%20>

“4 NETWORK CONFIGURATION IN CENTOS 8 | HOW TO CONFIGURE STATIC IP IN CENTOS 8 | Network Configuration”

<https://www.youtube.com/watch?v=tiUhtx-KzSc> for this source the language is a mix of Hindi and English

**Network Setup: Debian 11**

Network Interface configuration for Debian 11

Below is the output of the ifconfig command for Debian 11.

This will display network interface status in addition to ipv4 and ipv6 addresses for this virtual server.

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.177.129 netmask 255.255.255.0 broadcast 192.168.177.255

inet6 fe80::20c:29ff:fe08:1bda prefixlen 64 scopeid 0x20<link>

ether 00:0c:29:08:1b:da txqueuelen 1000 (Ethernet)

RX packets 46 bytes 5856 (5.7 KiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 48 bytes 4068 (3.9 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536

inet 127.0.0.1 netmask 255.0.0.0

inet6 ::1 prefixlen 128 scopeid 0x10<host>

loop txqueuelen 1000 (Local Loopback)

RX packets 8 bytes 400 (400.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 8 bytes 400 (400.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

Shown below is one of my configuration files for the debian server, this file shows the nameserver address.

Text

Description automatically generated

Network config file for Debian 11 setup file

Text

Description automatically generated

\*net-tools was already installed

on Debian 11

Installed programs Debian 11:

openssh-server 9/12/2021

Tmux installed 10/27/2021

Emacs installed 10/27/2021

Fail2Ban installed 10/27/2021

Cowsay installed 10/27/2021

Lolcat installed 10/27/2021

Vim installed 10/27/2021

Apt updated 10/27/2021

dnsutils installed 11/6/2021

**Network Setup: Centos 8**

Network interface configuration for Centos 8

Shown below is output for ifconfig command in Centos 8

Displayed is the network interfaces and their associated ipv4 and ipv6 addresses.

My script also runs this command I will include screenshots of the script output.

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.177.130 netmask 255.255.255.0 broadcast 192.168.177.255

inet6 fe80::20c:29ff:fe6d:b2e9 prefixlen 64 scopeid 0x20<link>

ether 00:0c:29:6d:b2:e9 txqueuelen 1000 (Ethernet)

RX packets 23115 bytes 34404074 (32.8 MiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 1859 bytes 130226 (127.1 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536

inet 127.0.0.1 netmask 255.0.0.0

inet6 ::1 prefixlen 128 scopeid 0x10<host>

loop txqueuelen 1000 (Local Loopback)

RX packets 4 bytes 240 (240.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 4 bytes 240 (240.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500

inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255

ether 52:54:00:f8:82:b4 txqueuelen 1000 (Ethernet)

RX packets 0 bytes 0 (0.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 0 bytes 0 (0.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

\*net-tools was already installed

on Centos 8

Configuration files

Network interface configuration file for centos 8 shown below

Text

Description automatically generated

Nameserver configuration file shown below for Centos 8

Text

Description automatically generated

Installed programs Centos 8:

dnf update Installed 10/29/2021

Emacs Installed 10/29/2021

Tmux Installed 10/29/2021

Epel-repository Installed 10/29/2021

Fail2ban Installed 10/29/2021

Cowsay Installed 10/29/2021

Ruby installed 10/29/21

Lolcat Installed 10/29/2021

Network Script and Research

For this lab I will be writing a script to dump network info to a text file.

Current Script

#!/bin/bash

#Nicholas Mueller Networking info dump script

#This script should be run in it's own directory it will create a file called netinfodump followed by the date and time.

#The script will append interface configuration info, listening port data, and dns data with the dig command to a file.

touch "netinfodump $(date)" #This line of the script creates the file to which data is appended along with the current date and time.

ifconfig >> "netinfodump $(date)" #First ifconfig is run its output is appended to the file previously created.

ss -l >> "netinfodump $(date)" #The ss command is like netstat it lists listening port status with the added -l option.

dig google.com >> "netinfodump $(date)" #Dig is run on google.com dig is a dns tool it provides the server address and dns info for google.com

Link to the script is here

<https://github.com/nmgithubuser2021/Portfolio-Linux-Admin/blob/main/Networkinfoscript.sh>

Instructions for running the script.

For Centos 8 this script will run fine using a non-root account, for Debian 11 the user must login as root. The networking script I have written is meant to be simple, nothing too fancy going on. As it is designed to run for multiple Linux operating systems, and I wanted it to be as automated as possible while still being effective and relatively simple. So, the idea behind this script is making it compatible across distros while being easy to use, but still effective at retrieving network data. This script should be run in its own directory I called mine networkinfo or something similar.

What does the Script do and why is it valuable

The first thing this script does is create a file with the date appended to the end of the file. I

accomplish this with the command touch “netinfodump then I use the date variable $(date) all

enclosed in quotation marks “. Now that I have a file with the date and purpose, the next step is to redirect command line output text into this file. The command ifconfig is run this command means interface configuration. Ifconfig running and saving the data in a file is valuable because, ifconfig displays ipv4 and ipv6 addresses. The ipv4 address I could use to sftp into the server to facilitate file transfer, additionally the interface’s status is shown along with packet data. All of which is useful information. If there is packet loss it indicates a problem with the network, so it is useful for administrators. Next command I have chosen to put in the script is ss -l command, this tool is similar to netstat but is newer and easier to use. The socket statistics command with the added -l option displays statistics for a wide range of listening sockets in a Linux machine. Various types of sockets such as TCP, Packets, UDP, etc will be shown. The ss command gets us valuable data on network services as well as associated connections and protocol statistics. So ss makes a valuable addition to the data collected that will be useful for troubleshooting network issues. The final command in the script is dig which stands for domain information groper, it is a dns lookup utility. Dig is similar to a replacement for the nslookup command, as socket statistics is for the netstat command. Dig is an administrative tool that is useful for the troubleshooting of DNS related name server problems. Dig will also display the ip address of a server including the server performing a query. It can also be used to query a server and return answers about that server including errors that may occur. Worth noting is that this command is not available in Debian 11 without installing the package dnsutils, it is available on Centos 8 without the dnsutils package.

Script Sample Run for Centos 8

The script is run with the command ./Scriptname.sh where Scriptname is the name of the script in this case the script is called Networkinfoscript.sh

Text

Description automatically generated

More script output is shown below

There are two more screenshots for centos followed by a screenshot of the script in vim the commands in the scripts are identical on my Debian 11 and Centos 8 servers.

Note some of the output for the script is not shown as it is very lengthy.

Text

Description automatically generated

Text

Description automatically generated

This is the contents of the script as shown in Centos 8 vim.Text

Description automatically generated

Script Sample Run for Debian 11

The script is run with the command ./Scriptname.sh where Scriptname is the name of the script, this is the output viewed in GNU nano

On my Debian server I display the output with nano. Using cat does not display well on the screen.Text

Description automatically generated

More output of the script shown below

Text

Description automatically generated

The last output from the script is shown below some data may not be shown in the screenshots as the output is very long

Text

Description automatically generated