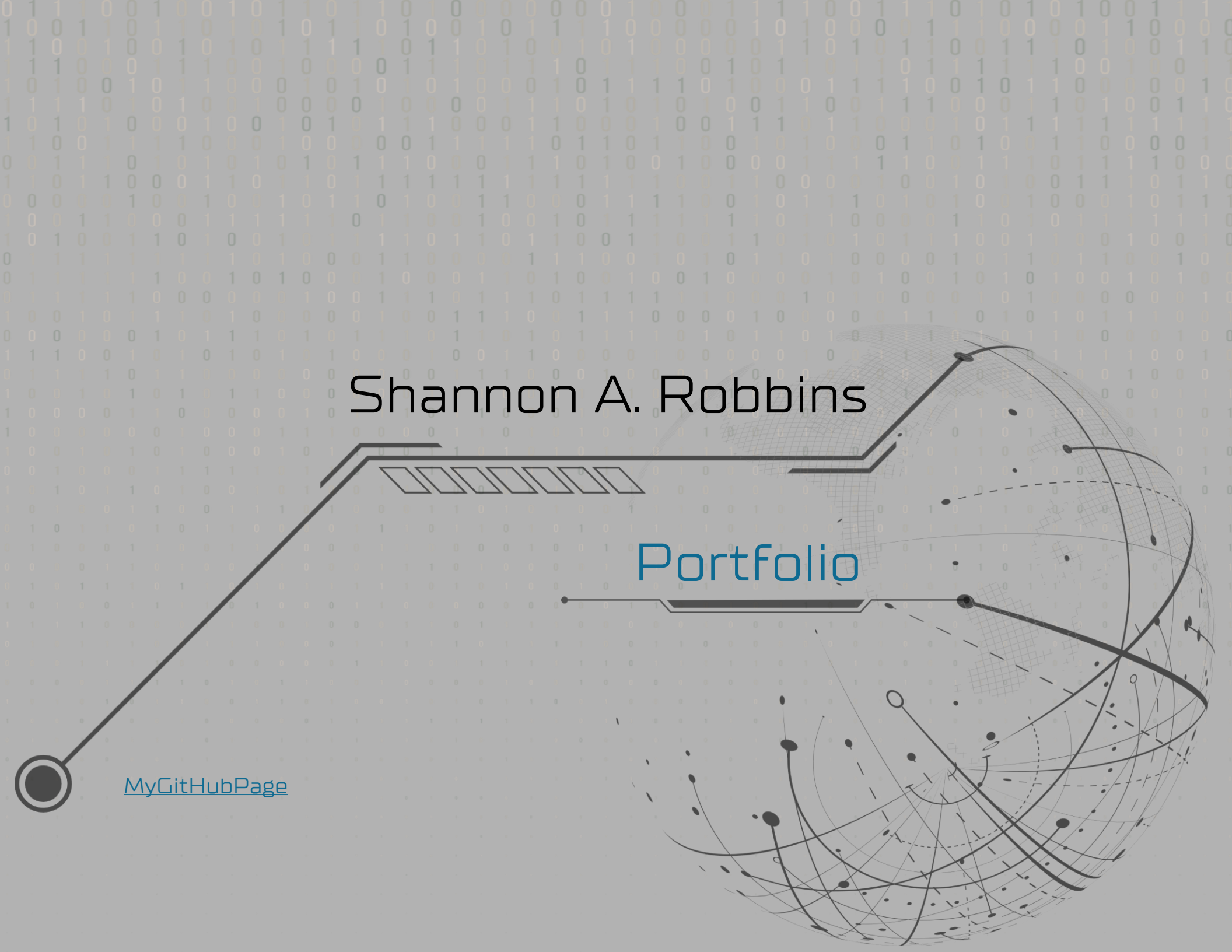
Shannon A. Robbins

Portfolio



MyGitHubPage

GrepLabLink

Grep Lab

grep -w ' Lane' GrepLab.txt

```
shanrobb@ubuntu:~/LinuxAdmin$ grep -w 'Lane' GrepLab.txt
Huckleberry Finn:238-923-7366:95 Latham Lane, Easton, PA 83755:11/12/56:20300
Betty Boop:245-836-8357:635 Cutesy Lane, Hollywood, CA 91464:6/23/23:14500
Lizzie Bennett:674-843-1385:20 Parak Lane, Duluth, MN 23850:4/12/23:780900
Nancy Drew:674-843-1385:20 Parak Lane, Duluth, MN 23850:4/12/23:780900
Dorothy Gale:923-835-8745:23 Wimp Lane, Kensington, DL 38758:8/31/69:126000
Molly Weasley:387-827-1095:13 Uno Lane, Ashville, NC 23556:7/1/29:57000
```

This command works by searching for the word "Lane." The "-w" option searches for the exact string between the quotes.

grep ^H GrepLab.txt

```
shanrobb@ubuntu:~/LinuxAdmin$ grep ^H GrepLab.txt
Huckleberry Finn:238-923-7366:95 Latham Lane, Easton, PA 83755:11/12/56:20300
Holly Golightly:397-857-2735:74 Pine Street, Dearborn, MI 23874:3/28/45:245700
Hester Prynne:408-253-3122:123 Park St., San Jose, CA 04086:7/25/53:85100
Hemione Granger:408-456-1234:4 Harvard Square, Boston, MA 02133:4/22/62:52600
```

This command matches lines within the document beginning with 'H' using '^' to do so.

GrepLabLink

Grep Lab

grep 000$ GrepLab.txt

```
shanrobb@ubuntu:~/LinuxAdmin$ grep 000$ GrepLab.txt
Meg Murry:834-938-8376:23445 Aster Ave., Allentown, NJ 83745:12/1/38:45000
Minerva McGonagall:408-233-8971:45 Rose Terrace, San Francisco, CA 92303:2/3/36:25000
Dorothy Gale:923-835-8745:23 Wimp Lane, Kensington, DL 38758:8/31/69:126000
Molly Weasley:387-827-1095:13 Uno Lane, Ashville, NC 23556:7/1/29:57000
```

By typing "$" after the string in your command, this will search the end of a line to find all lines that end with "000".

grep -v "408" GrepLab.txt

```
Westley Pirate:284-758-2867:23 Edgecliff Place, Lincoln, NB 92743:11/3/35:58200

Lizzie Bennett:674-843-1385:20 Parak Lane, Duluth, MN 23850:4/12/23:780900

Nancy Drew:674-843-1385:20 Parak Lane, Duluth, MN 23850:4/12/23:780900

Jo March:327-832-5728:3465 Mirlo Street, Peabody, MA 34756:10/2/65:35200

Victor Frankenstein:835-365-1284:454 Easy Street, Decatur, IL 75732:2/28/53:123500

Ephram Hardy:293-259-5395:235 CarltonLane, Joliet, IL 73858:8/12/20:56700

Meg Murry:834-938-8376:23445 Aster Ave., Allentown, NJ 83745:12/1/38:45000

Lucy Pevensie:385-573-8326:832 Ponce Drive, Gzary, IN 83756:12/1/46:268500
```

Using the "-v" option before your search word searches for all lines that do not contain it. This command will print every line that does not include "408" in it.

SedLabLink

Sed Lab

sed 's/Jo/Josephine/' SedLab

```
Josephine March:327-832-5728:3465 Mirlo Street, Peabody, MA 34756:10/2/65:35200

Victor Frankenstein:835-365-1284:454 Easy Street, Decatur, IL 75732:2/28/53:123500

Ephram Hardy:293-259-5395:235 CarltonLane, Josephineliet, IL 73858:8/12/20:56700
```

Using the "sed" command, outside of the quotes with "s" for substitution, allows me to use backslashes as delimiters to separate the original string ("Jo") with the new one that I am replacing it with ("Josephine") before closing the quotes and calling the file SedLab.

wc -l SedLab

```
shanrobb@ubuntu:~/LinuxAdmin$ wc -l SedLab
53 SedLab
```

The"wc -l" command will print the number of lines in the file to let the user know where to delete from.

sed '49, $d' SedLab

```
Dorothy Gale:923-835-8745:23 Wimp Lane, Kensington, DL 38758:8/31/69:126000

Popeye Sailor:156-454-3322:945 Bluto Street, NotHere, USA 29358:3/19/35:22350

Luna Lovegood:385-898-8357:38 Fife Way, Abilene, TX 39673:1/5/58:95600

shanrobb@ubuntu:~/LinuxAdmin$
```

With 53 lines, the number "49" is placed inside the quotes of this sed command, along with "$" - which represents the last line. Once executed, the command will know to "d" delete lines 49-53.

SedLabLink

Sed Lab

sed -n '3, 15 p' SedLab

```
shanrobb@ubuntu:~/LinuxAdmin$ sed -n '3, 15 p' SedLab
Betty Boop:245-836-8357:635 Cutesy Lane, Hollywood, CA 91464:6/23/23:14500

Dorian Gray:385-375-8395:3567 Populus Place, Caldwell, NJ 23875:6/18/68:23400

Holly Golightly:397-857-2735:74 Pine Street, Dearborn, MI 23874:3/28/45:245700

Ebenezer Scrooge:548-834-2348:583 Laurel Ave., Kingsville, TX 83745:10/1/35:58900

Hester Prynne:408-253-3122:123 Park St., San Jose, CA 04086:7/25/53:85100

Westley Pirate:284-758-2857:23 Edgecliff Place, Lincoln , NB 92743:7/25/53:85100

Westley Pirate:284-758-2867:23 Edgecliff Place, Lincoln, NB 92743:11/3/35:58200
```

In this sed command, "-n" allows the user to disable the automatic printing and only print lines 3 through 15 as was stated between the quotes along with "p".

sed '/CA/d' SedLab

```
Luna Lovegood:385-898-8357:38 Fife Way, Abilene, TX 39673:1/5/58:95600

Daenerys Targaryen:408-724-0140:1222 Oxbow Court, Sunnyvale, CA 94087:5/19/66:34200

Molly Weasley:387-827-1095:13 Uno Lane, Ashville, NC 23556:7/1/29:57000
```

```
Luna Lovegood:385-898-8357:38 Fife Way, Abilene, TX 39673:1/5/58:95600


Molly Weasley:387-827-1095:13 Uno Lane, Ashville, NC 23556:7/1/29:57000
```

Placing "/CA/d" between the quotes deletes all lines with the "CA" pattern deleting names from California.

AwkLabLink

Awk Lab

awk '{print $1}' AwkLab.data

```
shanrobb@ubuntu:~/LinuxAdmin$ awk '{print $1}' AwkLab.data
Samuel
Ponder
Angua
Susan
Tiffany
Sacharissa
Adora
Frodo
Tom
Peregrin
```

Using "$1" tells the command line to print the first field on each line. In this case, they are all first names.

awk -F ':' 'NR==8, NR==9 {print $1 $2}' AwkLab.data

```
shanrobb@ubuntu:~/LinuxAdmin$ awk -F ':' 'NR==8, NR==9 {print $1 $2}' AwkLab.data
Frodo Baggins(206) 548-1278
Tom Bombadil(916) 348-4278
```

In this command, the field separator (-F) is ":", and using this, the user can request line numbers 8 (NR==8) and 9 (NR==9) to print the first and second fields of those respective lines ($1, $2).

AwkLabLink

Awk Lab

awk -F ' ' 'NR==10 {print $1 $2}' AwkLab.data

```
shanrobb@ubuntu:~/LinuxAdmin$ awk -F ' ' 'NR==10 {print $1 $2}' AwkLab.data
PeregrinTook:(510)
```

This command allows the user to search for line number 10 (NR==10) while using a blank space ' ' as the field separator (-F) to print the first and second field – which happens to be Peregrin's first name and area code.

awk -F ':' '/123/{print $1 $2 $3}' AwkLab.data

```
shanrobb@ubuntu:~/LinuxAdmin$ awk -F ':' '/123/{print $1 $2 $3}' AwkLab.data
Antoine de Saint-Exupery(123) 978-6432250
Belladonna Took(123) 978-5754356
Eglantine Took(123) 978-3574473
```

Using slashes to specify the area code (123), the user can separate the fields (-F) 1, 2, and 3 ($1, $2, $3) by colons (:) and print the three entries from the inquiry.

CentOS_NetworkingScript

Networking Script

CentOS

Script to display network information on screen:

```
#!/bin/bash

# Variables
today=$(date +"%m-%d-%Y") # Today's date
ports=$(sudo ss -tunl) # Open ports
dns_serv=$(cat /etc/resolv.conf | grep nameserver) # DNS server
ipaddr=$(ip addr show) # IP address
rout_info=$(ip route) # Routing table

# New file to save network info
new_file="networkinfo_$today.txt" # Text file

# Displays network info
echo "Network Information - $today:" # Heading
echo "----------------------------" # Line separator
echo "IP Address:"
echo "$ipaddr" # Displays IP address info
echo "DNS Servers:"
echo "$dns_serv" # Displays DNS server info
echo "Open Ports:"
echo "$ports" # Displays info about open ports
echo "Routing Info:"
echo "$rout_info" # Displays routing info
```

Script to dump network information to file:

```
# Dumping network info
echo "Network Information - $today:" >> $new_file # Appends heading to the new file
echo "----------------------------" >> $new_file # Appends line separator to the new file
echo "IP Address:" >> $new_file
echo "$ipaddr" >> $new_file # Appends IP address info to the new file
echo "DNS Servers:" >> $new_file
echo "$dns_serv" >> $new_file # Appends DNS server info to the new file
echo "Open Ports:" >> $new_file
echo "$ports" >> $new_file # Appends port info to the new file
echo "Routing Info:" >> $new_file
echo "$rout_info" >> $new_file # Appends routing info to the new file
```

Networking Script

CentOS

When the script is run using *./networkinfo.sh*, the screen prints the following:

```
[shanrobb@CentOS ~]$ ./networkinfo.sh
Network Information: - 11-10-2024:
--------------------
IP Address:
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1
000
    link/ether 08:00:27:2a:38:ac brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global noprefixroute dynamic enp0s3
       valid_lft 85138sec preferred_lft 85138sec
    inet6 fe80::b80f:b71c:102:ca7d/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
DNS Servers:
nameserver 103.86.96.100
nameserver 103.86.99.100
nameserver 75.75.75.75
# NOTE: the libc resolver may not support more than 3 nameservers.
# The nameservers listed below may not be recognized.
nameserver 75.75.76.76
Open Ports:
Netid  State      Recv-Q Send-Q Local Address:Port               Peer Address:Port
udp    UNCONN     0      0      127.0.0.1:323                   *:*
udp    UNCONN     0      0         *:68                      *:*
udp    UNCONN     0      0         [::1]:323                  [::]:*
tcp    LISTEN     0      128       *:22                      *:*
tcp    LISTEN     0      100    127.0.0.1:25                      *:*
tcp    LISTEN     0      128    [::]:22                 [::]:*
tcp    LISTEN     0      100       [::1]:25                   [::]:*
Routing Info:
default via 10.0.2.2 dev enp0s3 proto dhcp metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15 metric 100
```

Ubuntu_NetworkingScript

Networking Script

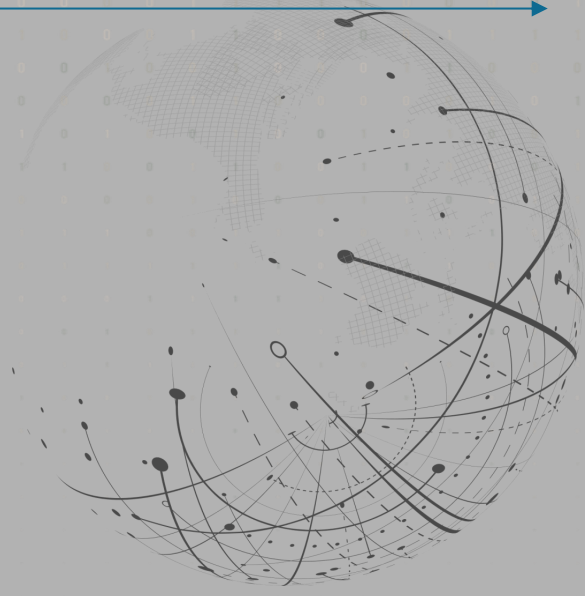
Ubuntu

Script to display network information on screen:

```
#!/bin/bash

# Variables
today=$(date +"%m-%d-%Y") # Today's date
ipaddr=$(ip addr show) # IP address
dns_serv=$(cat /etc/resolv.conf | grep nameserver) #DNS server
ports=$(sudo ss -tunl) # Open ports
rout_tbl=$(netstat -r) # Routing table
ns="NetworkSetup.txt" # Text file

# Variable for the file
net_file="network_info1_$today.txt"

# Displays network info
echo "Network Information - $today:" # Heading
echo "----------------------------" # Displays line separator
echo "IP Address:"
echo "$ipaddr" # Displays IP address info
echo "DNS Servers:"
echo "$dns_serv" # Displays DNS server info
echo "Open Ports:"
echo "$ports" # Displays info about open ports
echo "Routing Table:"
echo "$rout_tbl" # Displays routing info
```

Script to dump network information to file:

```
#Dumping network info
echo "Network Information - $today:" >> $net_file # Appends heading to the new file
echo "----------------------------" >> $net_file # Appends line separator to the new file
echo "IP Address:" >> $net_file
echo "$ipaddr" >> $net_file # Appends IP address info to the new file
echo "DNS Servers:" >> $net_file
echo "$dns_serv" >> $net_file # Appends DNS server info to the new file
echo "Open Ports:" >> $net_file
echo "$ports" >> $net_file # Appends port info to the new file
echo "Routing Table:" >> $net_file
echo "$rout_tbl" >> $net_file # Appends routing info to the new file
```



Networking Script

Ubuntu

When you run the script using *./networkinfo1.sh*:

```
Network Information - 11-17-2024:
---------------------------
IP Address:
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:fc:71:2c brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 86298sec preferred_lft 86298sec
    inet6 fe80::a00:27ff:fefc:712c/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default ql
en 1000
    link/ether 08:00:27:30:93:31 brd ff:ff:ff:ff:ff:ff
DNS Servers:
nameserver 127.0.0.53
Open Ports:
Netid State  Recv-Q Send-Q     Local Address:Port Peer Address:PortProcess
udp   UNCONN 0      0          127.0.0.53%lo:53         0.0.0.0:*
udp   UNCONN 0      0       10.0.2.15%enp0s3:68         0.0.0.0:*
tcp   LISTEN 0      4096       127.0.0.53%lo:53         0.0.0.0:*
tcp   LISTEN 0      128              0.0.0.0:22         0.0.0.0:*
tcp   LISTEN 0      128                 [::]:22            [::]:*
Routing Table:
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 enp0s3
10.0.2.0        0.0.0.0         255.255.255.0   U         0 0          0 enp0s3
_gateway        0.0.0.0         255.255.255.255 UH        0 0          0 enp0s3
cdns01.comcast. _gateway        255.255.255.255 UGH       0 0          0 enp0s3
cdns02.comcast. _gateway        255.255.255.255 UGH       0 0          0 enp0s3
103.86.96.100   _gateway        255.255.255.255 UGH       0 0          0 enp0s3
103.86.99.100   _gateway        255.255.255.255 UGH       0 0          0 enp0s3
```