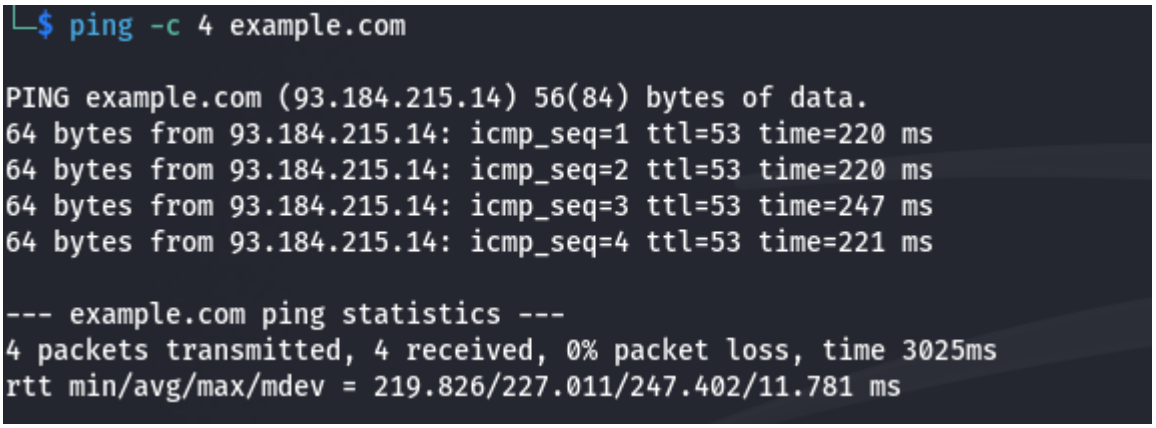
Lab 8 Assignment

Kruthi M S

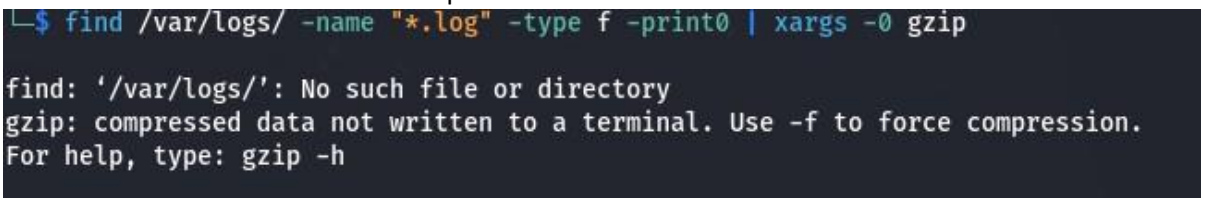
241059049

1. Use the ping command to test the connectivity to a remote server (e.g., example.com).

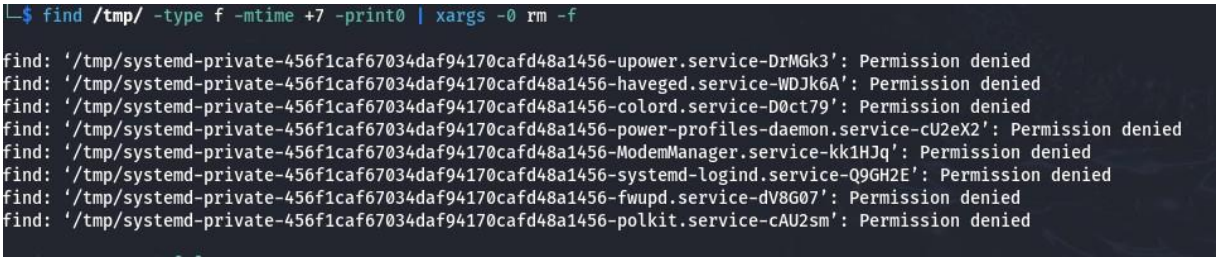


2. Write a script to measure the round-trip time for each packet and analyze the results.

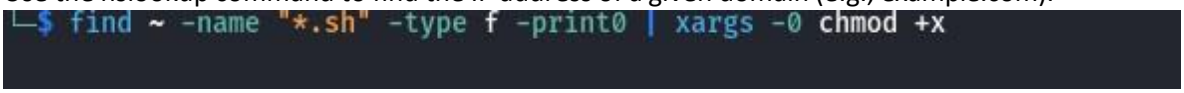
3. Use the traceroute to trace the route packets take to a destination



4. Analyze the output to identify any potential bottlenecks or points of failure in the route.



5. Use the nslookup command to find the IP address of a given domain (e.g., example.com).



6. Use the netstat command to view active connections and listening ports on your machine.

```
└─$ grep -i "auth" /etc/*.conf

/etc/ca-certificates.conf:mozilla/Actalis_Authentication_Root_CA.crt
/etc/ca-certificates.conf:mozilla/COMODO_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/COMODO_ECC_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/COMODO_RSA_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_EC1.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_G2.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_G4.crt
/etc/ca-certificates.conf:mozilla/ePKI_Root_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/GDCA_TrustAUTH_R5_ROOT.crt
/etc/ca-certificates.conf:mozilla/Go_Daddy_Root_Certificate_Authority_-_G2.crt
/etc/ca-certificates.conf:mozilla/Microsoft_ECC_Root_Certificate_Authority_2017.crt
/etc/ca-certificates.conf:mozilla/Microsoft_RSA_Root_Certificate_Authority_2017.crt
/etc/ca-certificates.conf:mozilla/NAVER_Global_Root_Certification_Authority.crt
```

7. Use the ifconfig (Linux) or ip a command to display network interface configurations.

```
└─$ grep -i "failed" /var/log/*.log | wc -l

grep: /var/log/boot.log: Permission denied
0
```

8. Write a script to report document the configuration of each interface, noting the IP address, subnet mask, and any other relevant information.

```
└─$ find . -maxdepth 1 -name "*.txt" -print0 | xargs -0 -I {} mv {} {}.bak
```

9. Perform a basic network scan using nmap on your local network to identify active devices and open ports.

```
└─$ #!/bin/bash
cat users.txt | xargs -I {} bash -c 'id -u {} &>/dev/null && echo "{} exists" || echo "{} does not exist"'

cat: users.txt: No such file or directory
```

10. Create a report summarizing the devices found, their IP addresses, and the services running on the open ports.

```
└─$ find /var/log -type f -size +1M -name "*.log" -print0 | xargs -0 grep -E "ERROR|CRITICAL"

find: ‘/var/log/private’: Permission denied
find: ‘/var/log/gdm3’: Permission denied
find: ‘/var/log/inetsim’: Permission denied
find: ‘/var/log/speech-dispatcher’: Permission denied
```

11. Capture network packets using tcpdump on a specific interface.

```
└─$ sudo tcpdump -i eth0 -w capture.pcap

[sudo] password for sanju:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes


^C0 packets captured
0 packets received by filter
0 packets dropped by kernel
```

12. Analyze the captured packets for specific protocols (like HTTP or DNS) and summarize your findings.

```
┌──(sanju㉿kali)-[~]
└─$ sudo tcpdump -r capture.pcap -nn -A 'port 80 or port 53'

reading from file capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
```

13. Use the whois command to gather registration information about a domain.

```
└─$ whois example.com

connect: Network is unreachable
```

14. Use the hostname command to display and change the hostname of your machine.

```
└─$ # Display hostname
hostname

# Change hostname (requires sudo privileges)
sudo hostnamectl set-hostname newhostname

kali
```

15. Use the finger command to gather information about users on a system.

```
┌──(kruthi㉿kali)-[~/Desktop]
└─$ finger
Login     Name       Tty      Idle  Login Time   Office     Office Phone
kruthi    kruthi     tty7     7:26  Nov  8 18:44 (:0)
```

16. Use the who command to see who is currently logged into the system and the last command to view the login history.

```
┌──(kruthi㉿kali)-[~/Desktop]
└─$ who
kruthi   tty7         2024-11-08 18:44 (:0)

┌──(kruthi㉿kali)-[~/Desktop]
└─$ last
kruthi   tty7         :0               Fri Nov  8 18:44    gone - no logout
reboot   system boot  6.6.15-amd64     Fri Nov  8 18:42   still running
kruthi   tty7         :0               Wed Nov  6 23:25 - crash (1+19:17)
reboot   system boot  6.6.15-amd64     Wed Nov  6 23:24   still running
kruthi   tty7         :0               Wed Nov  6 23:16 - crash  (00:08)
reboot   system boot  6.6.15-amd64     Wed Nov  6 23:16   still running
kruthi   tty7         :0               Wed Nov  6 20:49 - crash  (02:27)
reboot   system boot  6.6.15-amd64     Wed Nov  6 20:45   still running
kruthi   tty7         :0               Wed Nov  6 16:22 - crash  (04:22)
reboot   system boot  6.6.15-amd64     Wed Nov  6 16:22   still running
kruthi   tty7         :0               Wed Nov  6 15:14 - crash  (01:07)
reboot   system boot  6.6.15-amd64     Wed Nov  6 15:13   still running
kruthi   tty7         :0               Tue Nov  5 20:17 - crash  (18:55)
reboot   system boot  6.6.15-amd64     Tue Nov  5 20:17   still running
kruthi   tty7         :0               Tue Oct 29 23:12 - crash (6+21:04)
reboot   system boot  6.6.15-amd64     Tue Oct 29 23:11   still running
kruthi   tty7         :0               Thu Oct 24 17:33 - crash (5+05:38)
reboot   system boot  6.6.15-amd64     Thu Oct 24 17:32   still running
kruthi   tty7         :0               Fri Oct 18 02:05 - crash (6+15:26)
reboot   system boot  6.6.15-amd64     Fri Oct 18 02:05   still running
kruthi   tty7         :0               Thu Oct 10 09:30 - crash (7+16:35)
reboot   system boot  6.6.15-amd64     Thu Oct 10 09:29   still running
kruthi   tty7         :0               Wed Oct  9 19:18 - crash  (14:10)
reboot   system boot  6.6.15-amd64     Wed Oct  9 19:18   still running
```

Xargs

---

1. Write a shell script called testurl.sh that accepts a list of urls in a separate file and tests if the website is up or not.

```
└─$ #!/bin/bash
cat urls.txt | xargs -n 1 -I {} bash -c 'curl -Is {} | head -n 1 | grep -q "200" && echo "{} is up" || echo "{} is down"'

cat: urls.txt: No such file or directory
xargs: warning: options --max-args and --replace/-I/-i are mutually exclusive, ignoring previous --max-args value
```

2. Create a shell script called diskhog.sh that lists the 5 largest items (files or directories) in the current directory in decreasing order of size.

```
└─$ #!/bin/bash
du -ah . | sort -rh | head -n 5

154M    .
87M     ./.cache
62M     ./.cache/mozilla/firefox/dey9y6tm.default-esr
62M     ./.cache/mozilla/firefox
62M     ./.cache/mozilla
```

3. compress all .log files found in the /var/logs/ directory?

```
└─$ find /var/logs/ -name "*.log" -type f -print0 | xargs -0 gzip

find: '/var/logs/': No such file or directory
gzip: compressed data not written to a terminal. Use -f to force compression.
For help, type: gzip -h
```

4. delete all temporary files older than 7 days from the /tmp/ directory?

```
└─$ find /tmp/ -type f -mtime +7 -print0 | xargs -0 rm -f

find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-upower.service-DrMGk3': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-haveged.service-WDJk6A': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-colord.service-D0ct79': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-power-profiles-daemon.service-cU2eX2': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-ModemManager.service-kk1HJq': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-systemd-logind.service-Q9GH2E': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-fwupd.service-dV8G07': Permission denied
find: '/tmp/systemd-private-456f1caf67034daf94170cafd48a1456-polkit.service-cAU2sm': Permission denied
```

5. write a shell script to make all .sh files in your home directory executable?

```
└─$ find ~ -name "*.sh" -type f -print0 | xargs -0 chmod +x
```

6. search for the string "auth" in all .conf files in the /etc/ directory

```
└─$ grep -i "auth" /etc/*.conf

/etc/ca-certificates.conf:mozilla/Actalis_Authentication_Root_CA.crt
/etc/ca-certificates.conf:mozilla/COMODO_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/COMODO_ECC_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/COMODO_RSA_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_EC1.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_G2.crt
/etc/ca-certificates.conf:mozilla/Entrust_Root_Certification_Authority_-_G4.crt
/etc/ca-certificates.conf:mozilla/ePKI_Root_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/GDCA_TrustAUTH_R5_ROOT.crt
/etc/ca-certificates.conf:mozilla/Go_Daddy_Root_Certificate_Authority_-_G2.crt
/etc/ca-certificates.conf:mozilla/Microsoft_ECC_Root_Certificate_Authority_2017.crt
/etc/ca-certificates.conf:mozilla/Microsoft_RSA_Root_Certificate_Authority_2017.crt
/etc/ca-certificates.conf:mozilla/NAVER_Global_Root_Certification_Authority.crt
/etc/ca-certificates.conf:mozilla/Sectigo_Public_Server_Authentication_Root_E46.crt
/etc/ca-certificates.conf:mozilla/Sectigo_Public_Server_Authentication_Root_R46.crt
/etc/ca-certificates.conf:mozilla/SSL.com_EV_Root_Certification_Authority_ECC.crt
/etc/ca-certificates.conf:mozilla/SSL.com_EV_Root_Certification_Authority_RSA_R2.crt
```

7. count the number of "failed" login attempts in all .log files in /var/log/?

```
└─$ grep -i "failed" /var/log/*.log | wc -l

grep: /var/log/boot.log: Permission denied
0
```

8. rename all .txt files in the current directory by appending .bak

```
└─$ find . -maxdepth 1 -name "*.txt" -print0 | xargs -0 -I {} mv {} {}.bak
```

9. Write a shell script to check if a list of users from users.txt exist in the system.

```
└─$ #!/bin/bash
cat users.txt | xargs -I {} bash -c 'id -u {} &>/dev/null && echo "{} exists" || echo "{} does not exist"'

cat: users.txt: No such file or directory
```

10. search for keywords like "ERROR" or "CRITICAL" in all log files over 1MB in size.

```
└─$ find /var/log -type f -size +1M -name "*.log" -print0 | xargs -0 grep -E "ERROR|CRITICAL"

find: '/var/log/private': Permission denied
find: '/var/log/gdm3': Permission denied
find: '/var/log/inetsim': Permission denied
find: '/var/log/speech-dispatcher': Permission denied
```

==========

==========

"If everyone is moving forward together, then success takes care of itself." — Henry Ford