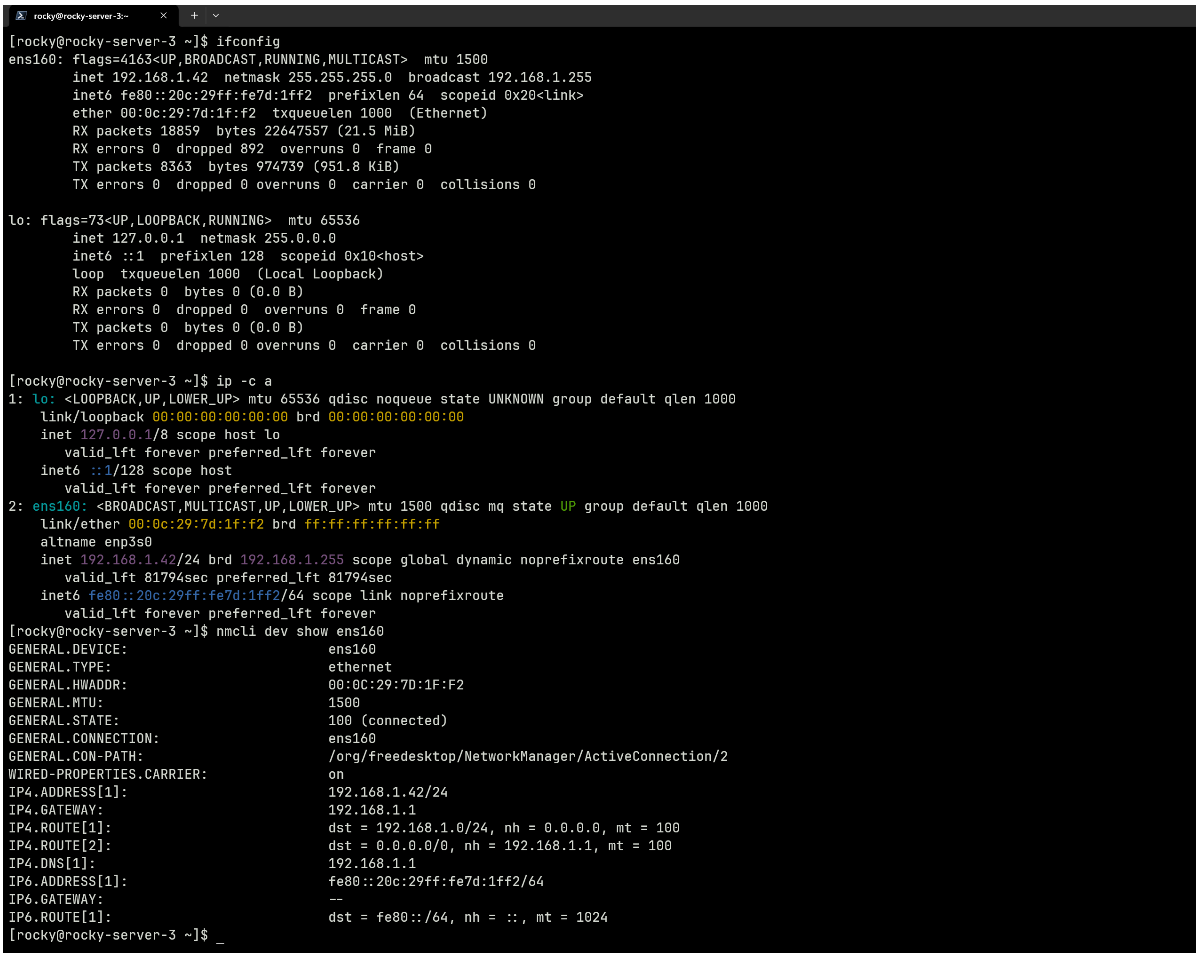
Activity 1

Checking the MAC address of the network card

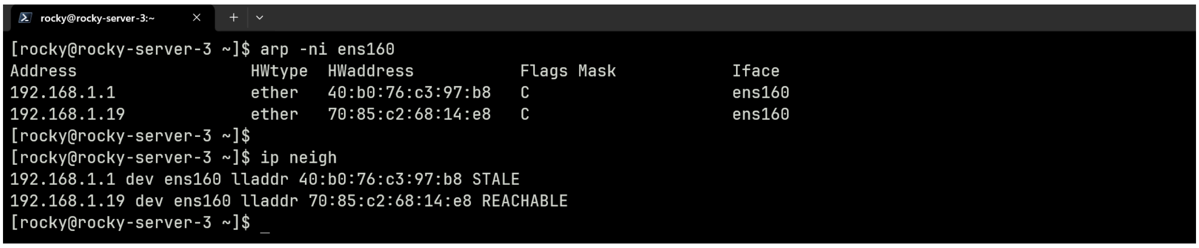
```
ifconfig

# -c flag colour the output,
# a flag means printing of all interfaces
ip -c a

# we can check the MAC address of a specific interface using the NetworkManager tool
nmcli dev show ens160

MAC address for ens160 interface is: 00:0c:29:7d:1f:f2
```



```
[rocky@rocky-server-3 ~]$ ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.42  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::20c:29ff:fe7d:1ff2  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:7d:1f:f2  txqueuelen 1000  (Ethernet)
        RX packets 18859  bytes 22647557 (21.5 MiB)
        RX errors 0  dropped 892  overruns 0  frame 0
        TX packets 8363  bytes 974739 (951.8 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 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

[rocky@rocky-server-3 ~]$ ip -c a
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: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:7d:1f:f2 brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 192.168.1.42/24 brd 192.168.1.255 scope global dynamic noprefixroute ens160
       valid_lft 81794sec preferred_lft 81794sec
    inet6 fe80::20c:29ff:fe7d:1ff2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
[rocky@rocky-server-3 ~]$ nmcli dev show ens160
GENERAL.DEVICE:                         ens160
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         00:0C:29:7D:1F:F2
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (connected)
GENERAL.CONNECTION:                     ens160
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/ActiveConnection/2
WIRED-PROPERTIES.CARRIER:               on
IP4.ADDRESS[1]:                         192.168.1.42/24
IP4.GATEWAY:                            192.168.1.1
IP4.ROUTE[1]:                           dst = 192.168.1.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh = 192.168.1.1, mt = 100
IP4.DNS[1]:                             192.168.1.1
IP6.ADDRESS[1]:                         fe80::20c:29ff:fe7d:1ff2/64
IP6.GATEWAY:                            --
IP6.ROUTE[1]:                           dst = fe80::/64, nh = ::, mt = 1024
[rocky@rocky-server-3 ~]$ _
```

Checking local ARP table

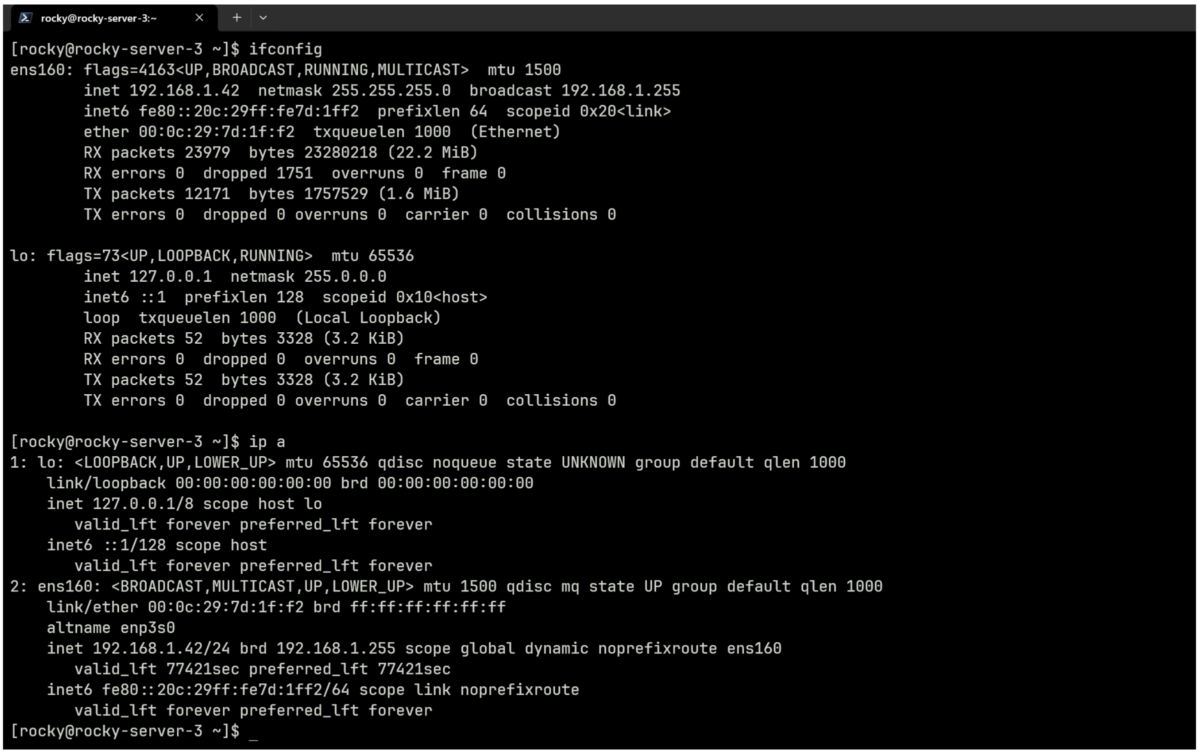
```
arp -ni ens160 # i flag allows us to select a specific interface, n flag show numerical addresses instead of symbolic host names
ip neigh
```



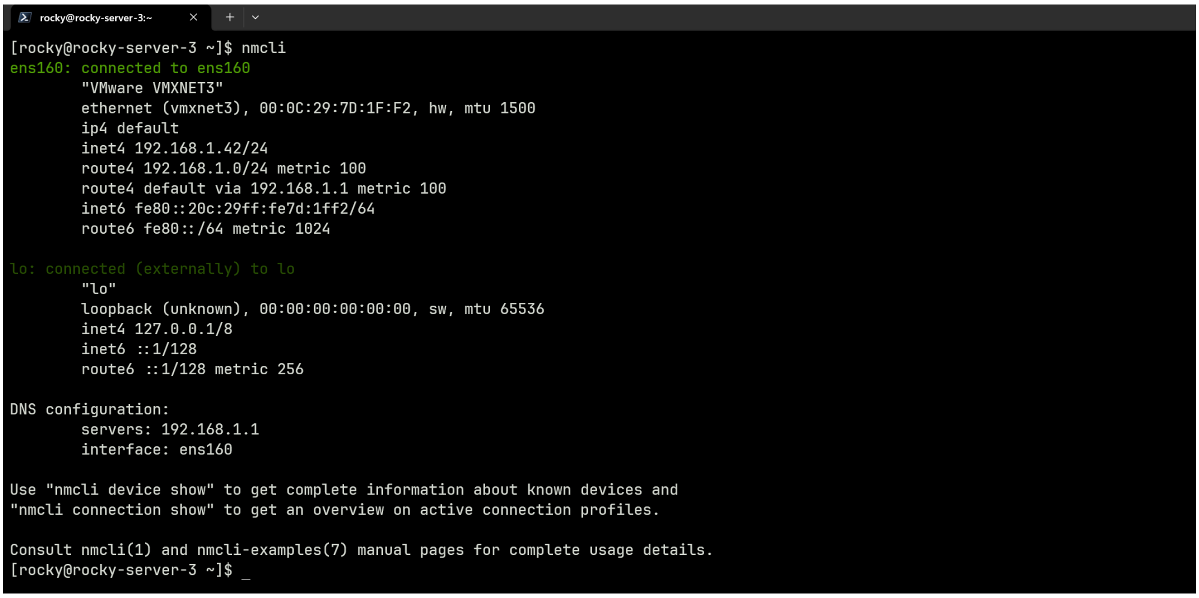
```
[rocky@rocky-server-3 ~]$ arp -ni ens160
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.1              ether   40:b0:76:c3:97:b8   C                     ens160
192.168.1.19             ether   70:85:c2:68:14:e8   C                     ens160
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ ip neigh
192.168.1.1 dev ens160 lladdr 40:b0:76:c3:97:b8 STALE
192.168.1.19 dev ens160 lladdr 70:85:c2:68:14:e8 REACHABLE
[rocky@rocky-server-3 ~]$ _
```

Checking network interfaces and associated IP addresses

```
ifconfig
ip a
nmcli
```



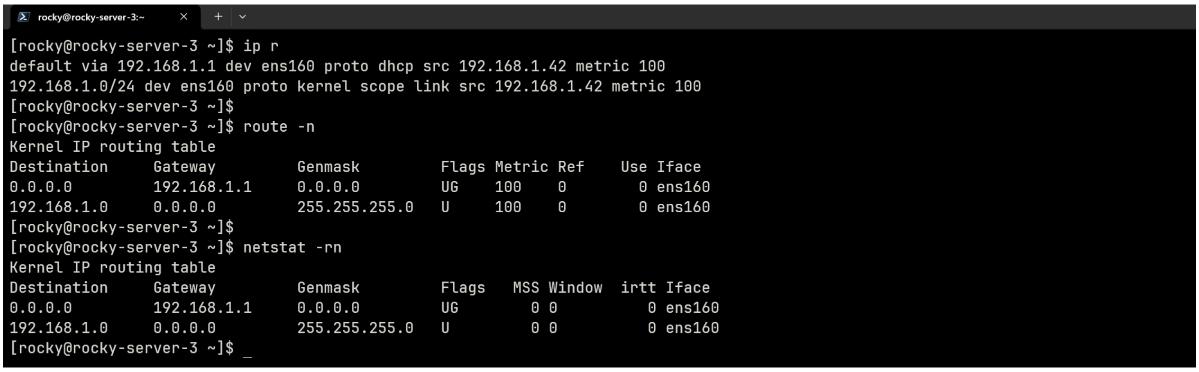
```
[rocky@rocky-server-3 ~]$ ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.42  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::20c:29ff:fe7d:1ff2  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:7d:1f:f2  txqueuelen 1000  (Ethernet)
        RX packets 23979  bytes 23280218 (22.2 MiB)
        RX errors 0  dropped 1751  overruns 0  frame 0
        TX packets 12171  bytes 1757529 (1.6 MiB)
        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 52  bytes 3328 (3.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 52  bytes 3328 (3.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[rocky@rocky-server-3 ~]$ ip a
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: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:7d:1f:f2 brd ff:ff:ff:ff:ff:ff
    altname enp3s0
    inet 192.168.1.42/24 brd 192.168.1.255 scope global dynamic noprefixroute ens160
       valid_lft 77421sec preferred_lft 77421sec
    inet6 fe80::20c:29ff:fe7d:1ff2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
[rocky@rocky-server-3 ~]$ _
```



```
[rocky@rocky-server-3 ~]$ nmcli
ens160: connected to ens160
        "VMware VMXNET3"
        ethernet (vmxnet3), 00:0C:29:7D:1F:F2, hw, mtu 1500
        ip4 default
        inet4 192.168.1.42/24
        route4 192.168.1.0/24 metric 100
        route4 default via 192.168.1.1 metric 100
        inet6 fe80::20c:29ff:fe7d:1ff2/64
        route6 fe80::/64 metric 1024

lo: connected (externally) to lo
        "lo"
        loopback (unknown), 00:00:00:00:00:00, sw, mtu 65536
        inet4 127.0.0.1/8
        inet6 ::1/128
        route6 ::1/128 metric 256

DNS configuration:
        servers: 192.168.1.1
        interface: ens160

Use "nmcli device show" to get complete information about known devices and
"nmcli connection show" to get an overview on active connection profiles.

Consult nmcli(1) and nmcli-examples(7) manual pages for complete usage details.
[rocky@rocky-server-3 ~]$ _
```

Checking the routing table

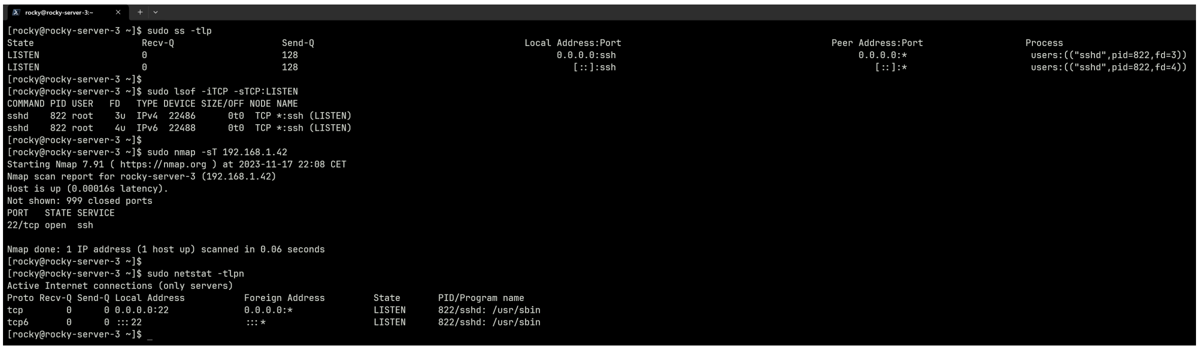
```
ip r # r flag shows table routes
route -n # n flag show numerical addresses instead of symbolic host names
netstat -rn # n flag show numerical addresses instead of symbolic host names, r flag shows routing tables
```



```
[rocky@rocky-server-3 ~]$ ip r
default via 192.168.1.1 dev ens160 proto dhcp src 192.168.1.42 metric 100
192.168.1.0/24 dev ens160 proto kernel scope link src 192.168.1.42 metric 100
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG    100    0        0 ens160
192.168.1.0     0.0.0.0         255.255.255.0   U     100    0        0 ens160
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG        0 0          0 ens160
192.168.1.0     0.0.0.0         255.255.255.0   U         0 0          0 ens160
[rocky@rocky-server-3 ~]$ _
```

Checking list of open (listening) TCP ports

```
sudo ss -tlp # t flag means TCP, l flag means listening ports, p flag means port
sudo lsof -iTCP -sTCP:LISTEN # List all TCP connections with state LISTEN
sudo nmap -sT 192.168.1.42 # -sT means TCP connect scan
sudo netstat -tlpn # t means TCP, l listening, p port, n show numerical addresses instead of symbolic host names
```

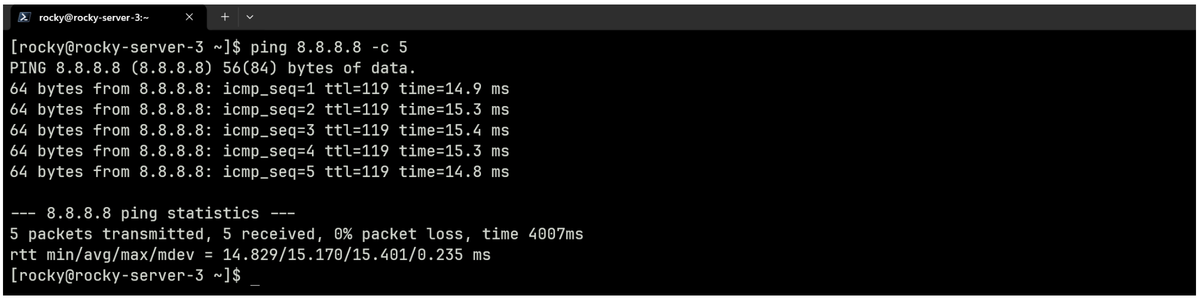


```
[rocky@rocky-server-3 ~]$ sudo ss -tlp
State                Recv-Q               Send-Q                              Local Address:Port                            Peer Address:Port               Process
LISTEN               0                    128                                       0.0.0.0:ssh                                  0.0.0.0:*                   users:(("sshd",pid=822,fd=3))
LISTEN               0                    128                                          [::]:ssh                                     [::]:*                   users:(("sshd",pid=822,fd=4))
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ sudo lsof -iTCP -sTCP:LISTEN
COMMAND PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
sshd    822 root    3u  IPv4  22486      0t0  TCP *:ssh (LISTEN)
sshd    822 root    4u  IPv6  22488      0t0  TCP *:ssh (LISTEN)
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ sudo nmap -sT 192.168.1.42
Starting Nmap 7.91 ( https://nmap.org ) at 2023-11-17 22:08 CET
Nmap scan report for rocky-server-3 (192.168.1.42)
Host is up (0.00016s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ sudo netstat -tlpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      822/sshd: /usr/sbin
tcp6       0      0 :::22                   :::*                    LISTEN      822/sshd: /usr/sbin
[rocky@rocky-server-3 ~]$ _
```

Activity 2

Subtask 1

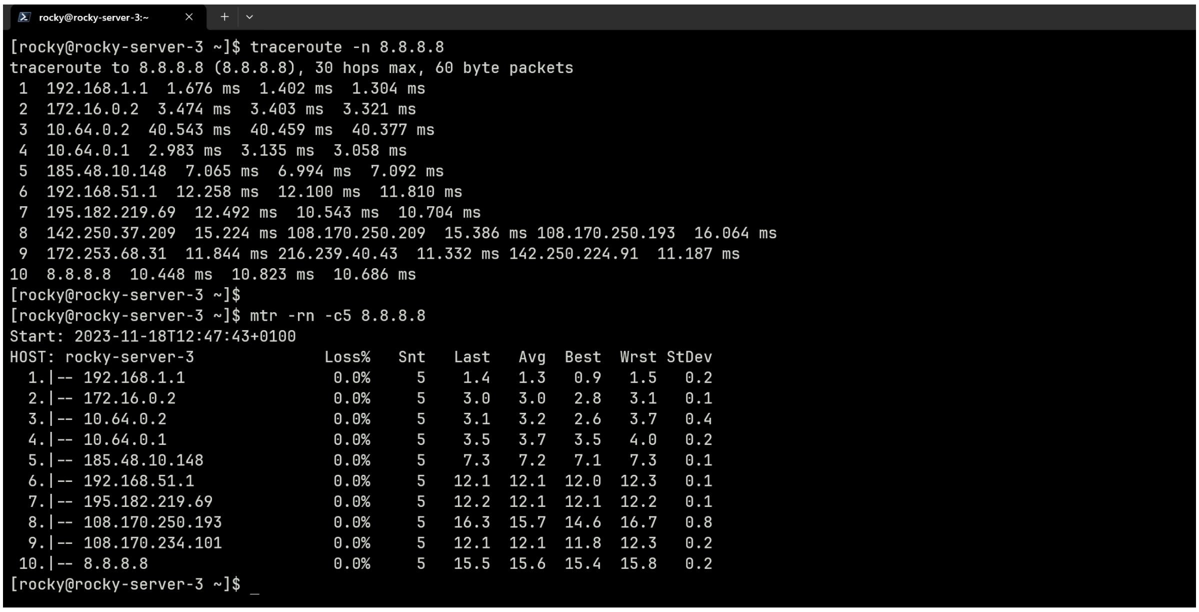
```
# c flag sending count ECHO_REQUEST packets.
ping 8.8.8.8 -c 5

# Output
# 0% packets loss
# RTT between my PC and google server (8.8.8.8) = min: 14.829, max: 15.401, avg: 15.170
```



```
[rocky@rocky-server-3 ~]$ ping 8.8.8.8 -c 5
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=14.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=15.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=15.4 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=119 time=15.3 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=119 time=14.8 ms

--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 14.829/15.170/15.401/0.235 ms
[rocky@rocky-server-3 ~]$ _
```

Subtask 2

```
traceroute -n 8.8.8.8 # n flag means that traceroute doesn't try to map IP addresses to host names.

mtr -rn -c5 8.8.8.8
# r flag puts mtr into report mode
# n flag display numeric IP numbers and not try to resolve the host names
# c flag set the number of pings sent to determine both the machines on the network and the reliability of those machines.
```



```
[rocky@rocky-server-3 ~]$ traceroute -n 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.1.1  1.676 ms  1.402 ms  1.304 ms
 2  172.16.0.2  3.474 ms  3.403 ms  3.321 ms
 3  10.64.0.2  40.543 ms  40.459 ms  40.377 ms
 4  10.64.0.1  2.983 ms  3.135 ms  3.058 ms
 5  185.48.10.148  7.065 ms  6.994 ms  7.092 ms
 6  192.168.51.1  12.258 ms  12.100 ms  11.810 ms
 7  195.182.219.69  12.492 ms  10.543 ms  10.704 ms
 8  142.250.37.209  15.224 ms 108.170.250.209  15.386 ms 108.170.250.193  16.064 ms
 9  172.253.68.31  11.844 ms 216.239.40.43  11.332 ms 142.250.224.91  11.187 ms
10  8.8.8.8  10.448 ms  10.823 ms  10.686 ms
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ mtr -rn -c5 8.8.8.8
Start: 2023-11-18T12:47:43+0100
HOST: rocky-server-3              Loss%   Snt   Last   Avg  Best  Wrst StDev
  1.|-- 192.168.1.1                0.0%     5    1.4   1.3   0.9   1.5   0.2
  2.|-- 172.16.0.2                 0.0%     5    3.0   3.0   2.8   3.1   0.1
  3.|-- 10.64.0.2                  0.0%     5    3.1   3.2   2.6   3.7   0.4
  4.|-- 10.64.0.1                  0.0%     5    3.5   3.7   3.5   4.0   0.2
  5.|-- 185.48.10.148              0.0%     5    7.3   7.2   7.1   7.3   0.1
  6.|-- 192.168.51.1               0.0%     5   12.1  12.1  12.0  12.3   0.1
  7.|-- 195.182.219.69             0.0%     5   12.2  12.1  12.1  12.2   0.1
  8.|-- 108.170.250.193            0.0%     5   16.3  15.7  14.6  16.7   0.8
  9.|-- 108.170.234.101            0.0%     5   12.1  12.1  11.8  12.3   0.2
 10.|-- 8.8.8.8                    0.0%     5   15.5  15.6  15.4  15.8   0.2
[rocky@rocky-server-3 ~]$ _
```

Activity 3

Subtask 1

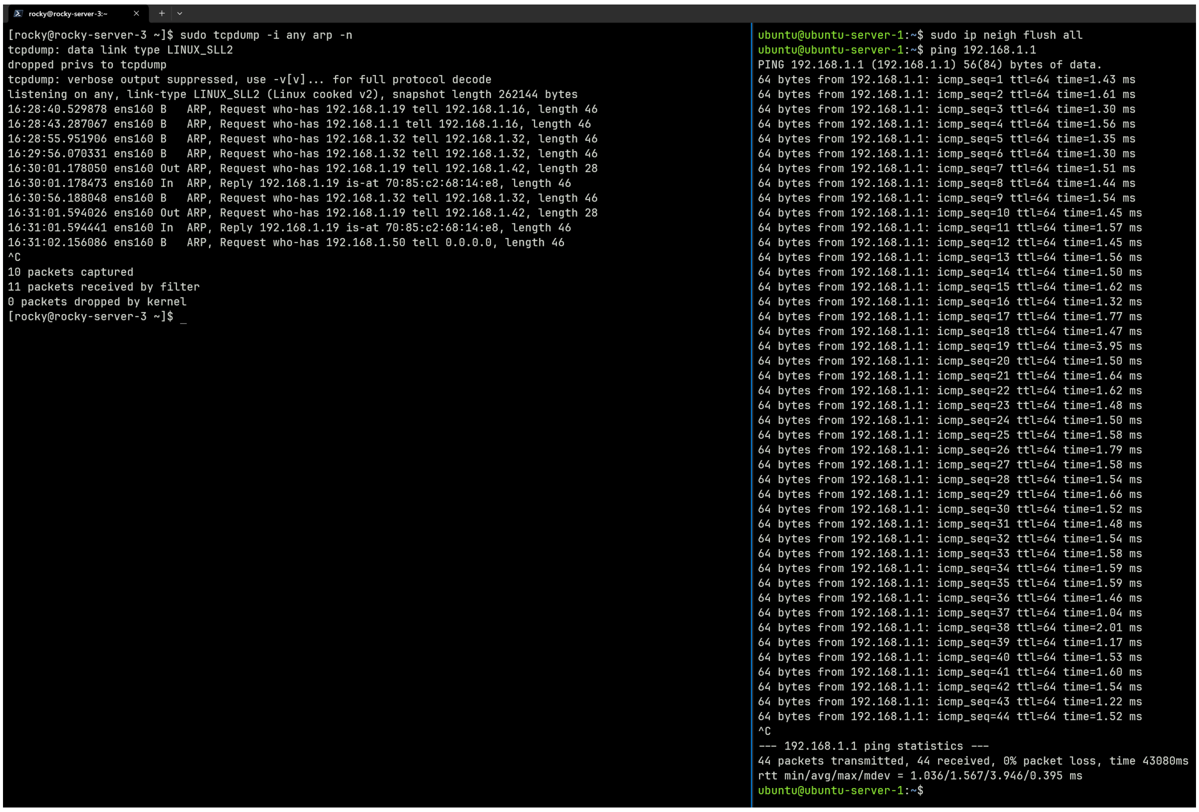
```
192.168.0.0/26
Netmask 255.255.255.192
Network address 192.168.0.0
Broadcast address 192.168.0.63
Number of hosts in the subnet: 62
```

1. **Netmask calculation:**
   CIDR 26
   192.168.0.0/26
   Each octet is 8 bits, CIDR is 26, it means that we need to calculate only the last octet
   11111111. 11111111. 11111111.?
   In 4th octet we left with two bits
   11111111. 11111111. 11111111.11000000
   Power of two: 128 64 32 16 8 4 2 1
   128 + 64 = 192
   NM 255.255.255.**192**
2. **Network address calculation:**
   192.168.0.0/26
   The network address in the network is the first one, so we fill it with zeros or we can follow the rules below:
   0 AND 0 = 0
   1 AND 0 = 0
   0 AND 1 = 0
   1 AND 1 = 1
   IP 192.168.0.0 = last octet is 0 0 0 0 0 0 0 0
   NA 192.168.0.0
3. **Broadcast address calculation:**
   192.168.0.0/26
   The broadcast address in the network is the last one, so we fill it with ones
   BA 192.168.0. 0 0 1 1 1 1 1 1 = 63
   BA 192.168.0.63
4. **Calculating the number of hosts in a subnet:**
   192.168.0.0/26
   IP address is 32-bit number. CIDR is 26, it means:
   32 - 26 = 6
   $2^6 = 64$
   We have a total of 64 addresses, but we also need to subtract the network address and the broadcast address.
   64 - 2 = 62
   Number of hosts in the subnet is **62**

Activity 4

Subtask 1

```
sudo tcpdump -i any arp -n # flag i means specific interface, flag n means to
disable DNS resolving  of hosts
sudo ip neigh flush all # clera local ARP cache
ping 192.168.1.1 # send an ICMP request to the default gateway (192.168.1.1)
```

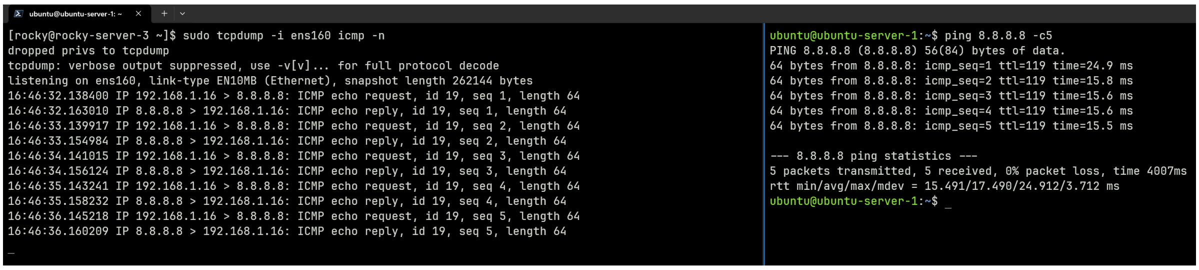


```
[rocky@rocky-server-3 ~]$ sudo tcpdump -i any arp -n
tcpdump: data link type LINUX_SLL2
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:28:40.529878 ens160 B   ARP, Request who-has 192.168.1.19 tell 192.168.1.16, length 46
16:28:43.287067 ens160 B   ARP, Request who-has 192.168.1.1 tell 192.168.1.16, length 46
16:28:55.951906 ens160 B   ARP, Request who-has 192.168.1.32 tell 192.168.1.32, length 46
16:29:56.070331 ens160 B   ARP, Request who-has 192.168.1.32 tell 192.168.1.32, length 46
16:30:01.178050 ens160 Out ARP, Request who-has 192.168.1.19 tell 192.168.1.42, length 28
16:30:01.178473 ens160 In  ARP, Reply 192.168.1.19 is-at 70:85:c2:68:14:e8, length 46
16:30:56.188048 ens160 B   ARP, Request who-has 192.168.1.32 tell 192.168.1.32, length 46
16:31:01.594026 ens160 Out ARP, Request who-has 192.168.1.19 tell 192.168.1.42, length 28
16:31:01.594441 ens160 In  ARP, Reply 192.168.1.19 is-at 70:85:c2:68:14:e8, length 46
16:31:02.156086 ens160 B   ARP, Request who-has 192.168.1.50 tell 0.0.0.0, length 46
^C
10 packets captured
11 packets received by filter
0 packets dropped by kernel
[rocky@rocky-server-3 ~]$ _
```

```
ubuntu@ubuntu-server-1:~$ sudo ip neigh flush all
ubuntu@ubuntu-server-1:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.43 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.61 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.30 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=1.56 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=1.35 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=1.30 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=64 time=1.51 ms
64 bytes from 192.168.1.1: icmp_seq=8 ttl=64 time=1.44 ms
64 bytes from 192.168.1.1: icmp_seq=9 ttl=64 time=1.54 ms
64 bytes from 192.168.1.1: icmp_seq=10 ttl=64 time=1.45 ms
64 bytes from 192.168.1.1: icmp_seq=11 ttl=64 time=1.57 ms
64 bytes from 192.168.1.1: icmp_seq=12 ttl=64 time=1.45 ms
64 bytes from 192.168.1.1: icmp_seq=13 ttl=64 time=1.56 ms
64 bytes from 192.168.1.1: icmp_seq=14 ttl=64 time=1.50 ms
64 bytes from 192.168.1.1: icmp_seq=15 ttl=64 time=1.62 ms
64 bytes from 192.168.1.1: icmp_seq=16 ttl=64 time=1.32 ms
64 bytes from 192.168.1.1: icmp_seq=17 ttl=64 time=1.77 ms
64 bytes from 192.168.1.1: icmp_seq=18 ttl=64 time=1.47 ms
64 bytes from 192.168.1.1: icmp_seq=19 ttl=64 time=3.95 ms
64 bytes from 192.168.1.1: icmp_seq=20 ttl=64 time=1.50 ms
64 bytes from 192.168.1.1: icmp_seq=21 ttl=64 time=1.64 ms
64 bytes from 192.168.1.1: icmp_seq=22 ttl=64 time=1.62 ms
64 bytes from 192.168.1.1: icmp_seq=23 ttl=64 time=1.48 ms
64 bytes from 192.168.1.1: icmp_seq=24 ttl=64 time=1.50 ms
64 bytes from 192.168.1.1: icmp_seq=25 ttl=64 time=1.58 ms
64 bytes from 192.168.1.1: icmp_seq=26 ttl=64 time=1.79 ms
64 bytes from 192.168.1.1: icmp_seq=27 ttl=64 time=1.58 ms
64 bytes from 192.168.1.1: icmp_seq=28 ttl=64 time=1.54 ms
64 bytes from 192.168.1.1: icmp_seq=29 ttl=64 time=1.66 ms
64 bytes from 192.168.1.1: icmp_seq=30 ttl=64 time=1.52 ms
64 bytes from 192.168.1.1: icmp_seq=31 ttl=64 time=1.48 ms
64 bytes from 192.168.1.1: icmp_seq=32 ttl=64 time=1.54 ms
64 bytes from 192.168.1.1: icmp_seq=33 ttl=64 time=1.58 ms
64 bytes from 192.168.1.1: icmp_seq=34 ttl=64 time=1.59 ms
64 bytes from 192.168.1.1: icmp_seq=35 ttl=64 time=1.59 ms
64 bytes from 192.168.1.1: icmp_seq=36 ttl=64 time=1.46 ms
64 bytes from 192.168.1.1: icmp_seq=37 ttl=64 time=1.04 ms
64 bytes from 192.168.1.1: icmp_seq=38 ttl=64 time=2.01 ms
64 bytes from 192.168.1.1: icmp_seq=39 ttl=64 time=1.17 ms
64 bytes from 192.168.1.1: icmp_seq=40 ttl=64 time=1.53 ms
64 bytes from 192.168.1.1: icmp_seq=41 ttl=64 time=1.60 ms
64 bytes from 192.168.1.1: icmp_seq=42 ttl=64 time=1.54 ms
64 bytes from 192.168.1.1: icmp_seq=43 ttl=64 time=1.22 ms
64 bytes from 192.168.1.1: icmp_seq=44 ttl=64 time=1.52 ms
^C
--- 192.168.1.1 ping statistics ---
44 packets transmitted, 44 received, 0% packet loss, time 43080ms
rtt min/avg/max/mdev = 1.036/1.567/3.946/0.395 ms
ubuntu@ubuntu-server-1:~$
```

Subtask 2

```
sudo tcpdump -i ens160 icmp -n
ping 8.8.8.8 -c5 # send 5 ICMP request to 8.8.8.8
```

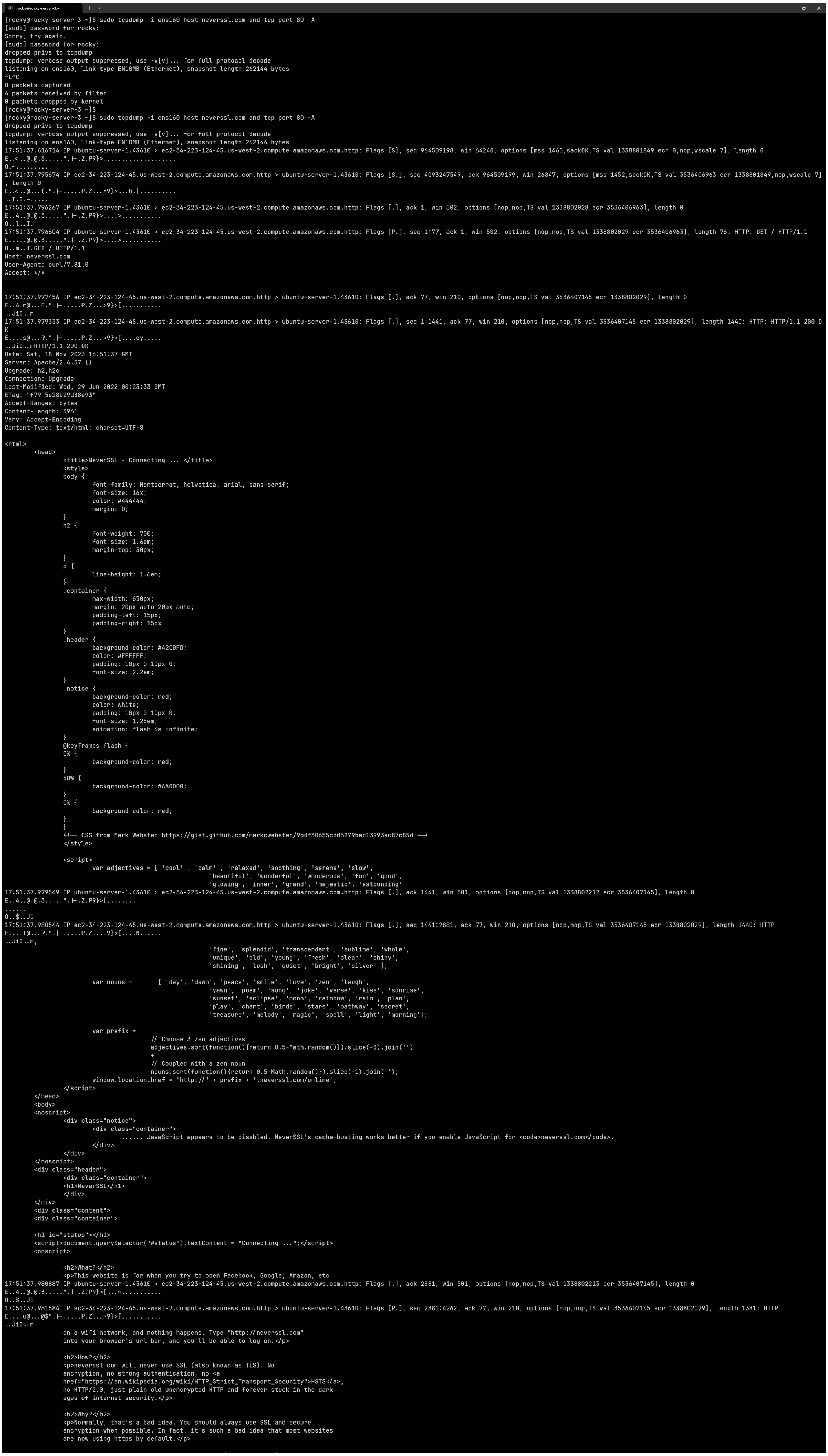


```
[rocky@rocky-server-3 ~]$ sudo tcpdump -i ens160 icmp -n
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens160, link-type EN10MB (Ethernet), snapshot length 262144 bytes
16:46:32.138400 IP 192.168.1.16 > 8.8.8.8: ICMP echo request, id 19, seq 1, length 64
16:46:32.163010 IP 8.8.8.8 > 192.168.1.16: ICMP echo reply, id 19, seq 1, length 64
16:46:33.139917 IP 192.168.1.16 > 8.8.8.8: ICMP echo request, id 19, seq 2, length 64
16:46:33.154984 IP 8.8.8.8 > 192.168.1.16: ICMP echo reply, id 19, seq 2, length 64
16:46:34.141015 IP 192.168.1.16 > 8.8.8.8: ICMP echo request, id 19, seq 3, length 64
16:46:34.156124 IP 8.8.8.8 > 192.168.1.16: ICMP echo reply, id 19, seq 3, length 64
16:46:35.143241 IP 192.168.1.16 > 8.8.8.8: ICMP echo request, id 19, seq 4, length 64
16:46:35.158232 IP 8.8.8.8 > 192.168.1.16: ICMP echo reply, id 19, seq 4, length 64
16:46:36.145218 IP 192.168.1.16 > 8.8.8.8: ICMP echo request, id 19, seq 5, length 64
16:46:36.160209 IP 8.8.8.8 > 192.168.1.16: ICMP echo reply, id 19, seq 5, length 64

_
```

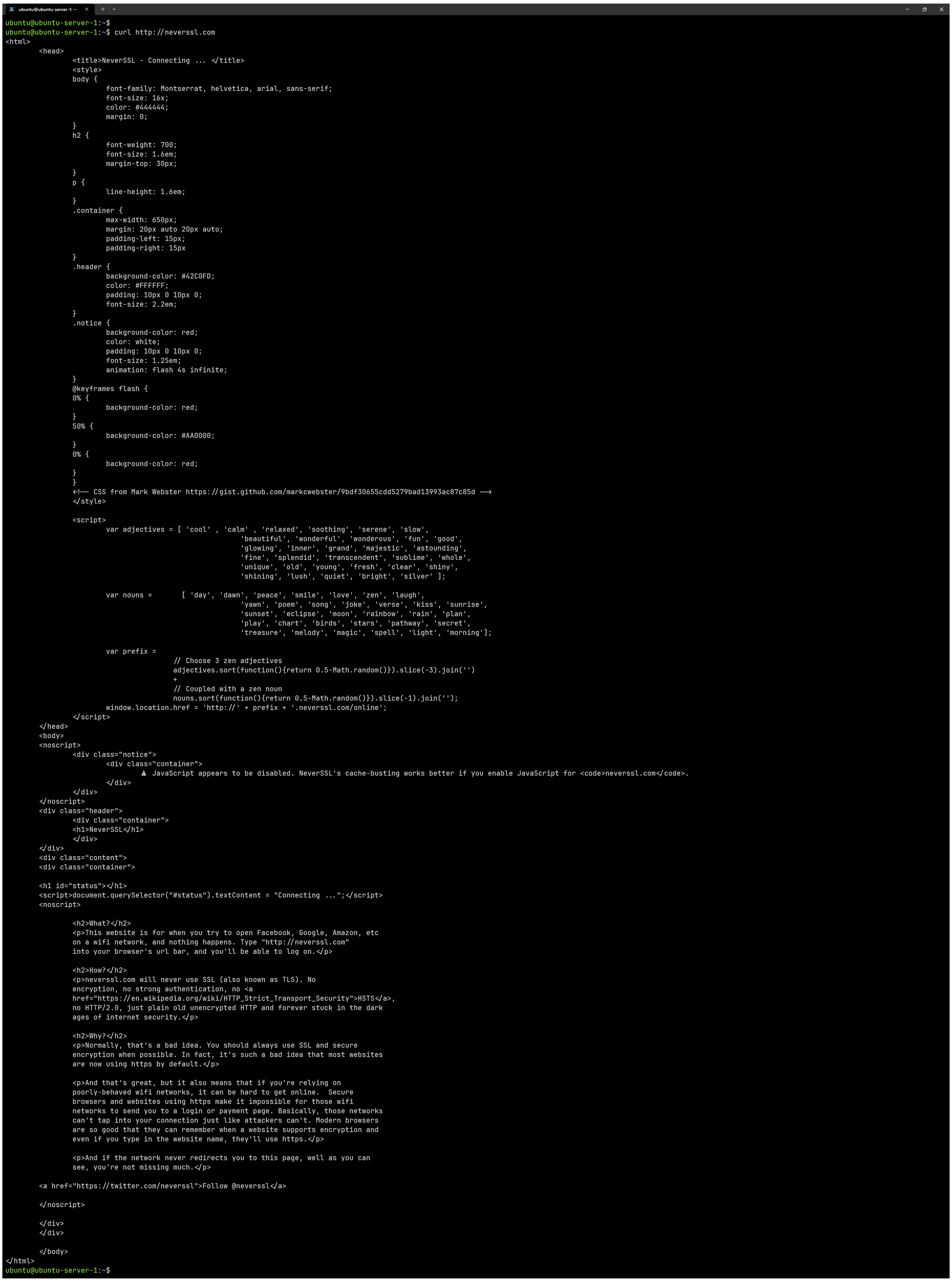
```
ubuntu@ubuntu-server-1:~$ ping 8.8.8.8 -c5
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=24.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=15.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=15.6 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=119 time=15.6 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=119 time=15.5 ms

--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 15.491/17.490/24.912/3.712 ms
ubuntu@ubuntu-server-1:~$ _
```

Subtask 3

```
sudo tcpdump -i ens160 host neverssl.com and tcp port 80 -A
# flag A display captured packets in ASCII

# curl tool was used to send HTTP request to neverssl.com
curl http://neverssl.com
```



```
[rocky@rocky-server-3 ~]$ sudo tcpdump -i ens160 host neverssl.com and tcp port 80 -A
[sudo] password for rocky:
Sorry, try again.
[sudo] password for rocky:
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens160, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^L^C
0 packets captured
4 packets received by filter
0 packets dropped by kernel
[rocky@rocky-server-3 ~]$
[rocky@rocky-server-3 ~]$ sudo tcpdump -i ens160 host neverssl.com and tcp port 80 -A
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens160, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:51:37.616714 IP ubuntu-server-1.43610 > ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http: Flags [S], seq 964509198, win 64240, options [mss 1460,sackOK,TS val 1338801849 ecr 0,nop,wscale 7], length 0
E..<..@.@.3.....".⊢.Z.P9}>...................
O.~.........
17:51:37.795674 IP ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http > ubuntu-server-1.43610: Flags [S.], seq 4093247549, ack 964509199, win 26847, options [mss 1452,sackOK,TS val 3536406963 ecr 1338801849,nop,wscale 7]
, length 0
E..<..@...(.".⊢.....P.Z...=9}>...h.|.........
..I.O.~.....
17:51:37.796267 IP ubuntu-server-1.43610 > ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http: Flags [.], ack 1, win 502, options [nop,nop,TS val 1338802028 ecr 3536406963], length 0
E..4..@.@.3.....".⊢.Z.P9}>....>..........
O..l..I.
17:51:37.796604 IP ubuntu-server-1.43610 > ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http: Flags [P.], seq 1:77, ack 1, win 502, options [nop,nop,TS val 1338802029 ecr 3536406963], length 76: HTTP: GET / HTTP/1.1
E.....@.@.3.....".⊢.Z.P9}>....>..........
O..m..I.GET / HTTP/1.1
Host: neverssl.com
User-Agent: curl/7.81.0
Accept: */*


17:51:37.977456 IP ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http > ubuntu-server-1.43610: Flags [.], ack 77, win 210, options [nop,nop,TS val 3536407145 ecr 1338802029], length 0
E..4.r@...E.".⊢.....P.Z...>9}>[..........
..JiO..m
17:51:37.979333 IP ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http > ubuntu-server-1.43610: Flags [.], seq 1:1441, ack 77, win 210, options [nop,nop,TS val 3536407145 ecr 1338802029], length 1440: HTTP: HTTP/1.1 200 O
K
E....s@...?.".⊢.....P.Z...>9}>[....ey.....
..JiO..mHTTP/1.1 200 OK
Date: Sat, 18 Nov 2023 16:51:37 GMT
Server: Apache/2.4.57 ()
Upgrade: h2,h2c
Connection: Upgrade
Last-Modified: Wed, 29 Jun 2022 00:23:33 GMT
ETag: "f79-5e28b29d38e93"
Accept-Ranges: bytes
Content-Length: 3961
Vary: Accept-Encoding
Content-Type: text/html; charset=UTF-8

<html>
        <head>
                <title>NeverSSL - Connecting ... </title>
                <style>
                body {
                        font-family: Montserrat, helvetica, arial, sans-serif;
                        font-size: 16x;
                        color: #444444;
                        margin: 0;
                }
                h2 {
                        font-weight: 700;
                        font-size: 1.6em;
                        margin-top: 30px;
                }
                p {
                        line-height: 1.6em;
                }
                .container {
                        max-width: 650px;
                        margin: 20px auto 20px auto;
                        padding-left: 15px;
                        padding-right: 15px
                }
                .header {
                        background-color: #42C0FD;
                        color: #FFFFFF;
                        padding: 10px 0 10px 0;
                        font-size: 2.2em;
                }
                .notice {
                        background-color: red;
                        color: white;
                        padding: 10px 0 10px 0;
                        font-size: 1.25em;
                        animation: flash 4s infinite;
                }
                @keyframes flash {
                0% {
                        background-color: red;
                }
                50% {
                        background-color: #AA0000;
                }
                0% {
                        background-color: red;
                }
                }
                <!-- CSS from Mark Webster https://gist.github.com/markcwebster/9bdf30655cdd5279bad13993ac87c85d -->
                </style>

                <script>
                        var adjectives = [ 'cool' , 'calm' , 'relaxed', 'soothing', 'serene', 'slow',
                                                        'beautiful', 'wonderful', 'wonderous', 'fun', 'good',
                                                        'glowing', 'inner', 'grand', 'majestic', 'astounding'
17:51:37.979549 IP ubuntu-server-1.43610 > ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http: Flags [.], ack 1441, win 501, options [nop,nop,TS val 1338802212 ecr 3536407145], length 0
E..4..@.@.3.....".⊢.Z.P9}>[........
......
O..$..Ji
17:51:37.980544 IP ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http > ubuntu-server-1.43610: Flags [.], seq 1441:2881, ack 77, win 210, options [nop,nop,TS val 3536407145 ecr 1338802029], length 1440: HTTP
E....t@...?.".⊢.....P.Z....9}>[....N......
..JiO..m,
                                                        'fine', 'splendid', 'transcendent', 'sublime', 'whole',
                                                        'unique', 'old', 'young', 'fresh', 'clear', 'shiny',
                                                        'shining', 'lush', 'quiet', 'bright', 'silver' ];

                        var nouns =       [ 'day', 'dawn', 'peace', 'smile', 'love', 'zen', 'laugh',
                                                        'yawn', 'poem', 'song', 'joke', 'verse', 'kiss', 'sunrise',
                                                        'sunset', 'eclipse', 'moon', 'rainbow', 'rain', 'plan',
                                                        'play', 'chart', 'birds', 'stars', 'pathway', 'secret',
                                                        'treasure', 'melody', 'magic', 'spell', 'light', 'morning'];

                        var prefix =
                                        // Choose 3 zen adjectives
                                        adjectives.sort(function(){return 0.5-Math.random()}).slice(-3).join('')
                                        +
                                        // Coupled with a zen noun
                                        nouns.sort(function(){return 0.5-Math.random()}).slice(-1).join('');
                        window.location.href = 'http://' + prefix + '.neverssl.com/online';
                </script>
        </head>
        <body>
        <noscript>
                <div class="notice">
                        <div class="container">
                                ...... JavaScript appears to be disabled. NeverSSL's cache-busting works better if you enable JavaScript for <code>neverssl.com</code>.
                        </div>
                </div>
        </noscript>
        <div class="header">
                <div class="container">
                <h1>NeverSSL</h1>
                </div>
        </div>
        <div class="content">
        <div class="container">

        <h1 id="status"></h1>
        <script>document.querySelector("#status").textContent = "Connecting ...";</script>
        <noscript>

                <h2>What?</h2>
                <p>This website is for when you try to open Facebook, Google, Amazon, etc
17:51:37.980887 IP ubuntu-server-1.43610 > ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http: Flags [.], ack 2881, win 501, options [nop,nop,TS val 1338802213 ecr 3536407145], length 0
E..4..@.@.3.....".⊢.Z.P9}>[...~..........
O..%..Ji
17:51:37.981584 IP ec2-34-223-124-45.us-west-2.compute.amazonaws.com.http > ubuntu-server-1.43610: Flags [P.], seq 2881:4262, ack 77, win 210, options [nop,nop,TS val 3536407145 ecr 1338802029], length 1381: HTTP
E....u@...@$".⊢.....P.Z...~9}>[..........
..JiO..m
                on a wifi network, and nothing happens. Type "http://neverssl.com"
                into your browser's url bar, and you'll be able to log on.</p>

                <h2>How?</h2>
                <p>neverssl.com will never use SSL (also known as TLS). No
                encryption, no strong authentication, no <a
                href="https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security">HSTS</a>,
                no HTTP/2.0, just plain old unencrypted HTTP and forever stuck in the dark
                ages of internet security.</p>

                <h2>Why?</h2>
                <p>Normally, that's a bad idea. You should always use SSL and secure
                encryption when possible. In fact, it's such a bad idea that most websites
                are now using https by default.</p>
```



```
ubuntu@ubuntu-server-1:~$
ubuntu@ubuntu-server-1:~$ curl http://neverssl.com
<html>
        <head>
                <title>NeverSSL - Connecting ... </title>
                <style>
                body {
                        font-family: Montserrat, helvetica, arial, sans-serif;
                        font-size: 16x;
                        color: #444444;
                        margin: 0;
                }
                h2 {
                        font-weight: 700;
                        font-size: 1.6em;
                        margin-top: 30px;
                }
                p {
                        line-height: 1.6em;
                }
                .container {
                        max-width: 650px;
                        margin: 20px auto 20px auto;
                        padding-left: 15px;
                        padding-right: 15px
                }
                .header {
                        background-color: #42C0FD;
                        color: #FFFFFF;
                        padding: 10px 0 10px 0;
                        font-size: 2.2em;
                }
                .notice {
                        background-color: red;
                        color: white;
                        padding: 10px 0 10px 0;
                        font-size: 1.25em;
                        animation: flash 4s infinite;
                }
                @keyframes flash {
                0% {
                        background-color: red;
                }
                50% {
                        background-color: #AA0000;
                }
                0% {
                        background-color: red;
                }
                }
                <!-- CSS from Mark Webster https://gist.github.com/markcwebster/9bdf30655cdd5279bad13993ac87c85d -->
                </style>

                <script>
                        var adjectives = [ 'cool' , 'calm' , 'relaxed', 'soothing', 'serene', 'slow',
                                                'beautiful', 'wonderful', 'wonderous', 'fun', 'good',
                                                'glowing', 'inner', 'grand', 'majestic', 'astounding',
                                                'fine', 'splendid', 'transcendent', 'sublime', 'whole',
                                                'unique', 'old', 'young', 'fresh', 'clear', 'shiny',
                                                'shining', 'lush', 'quiet', 'bright', 'silver' ];

                        var nouns =       [ 'day', 'dawn', 'peace', 'smile', 'love', 'zen', 'laugh',
                                                'yawn', 'poem', 'song', 'joke', 'verse', 'kiss', 'sunrise',
                                                'sunset', 'eclipse', 'moon', 'rainbow', 'rain', 'plan',
                                                'play', 'chart', 'birds', 'stars', 'pathway', 'secret',
                                                'treasure', 'melody', 'magic', 'spell', 'light', 'morning'];

                        var prefix =
                                        // Choose 3 zen adjectives
                                        adjectives.sort(function(){return 0.5-Math.random()}).slice(-3).join('')
                                        +
                                        // Coupled with a zen noun
                                        nouns.sort(function(){return 0.5-Math.random()}).slice(-1).join('');
                        window.location.href = 'http://' + prefix + '.neverssl.com/online';
                </script>
        </head>
        <body>
        <noscript>
                <div class="notice">
                        <div class="container">
                                ⚠ JavaScript appears to be disabled. NeverSSL's cache-busting works better if you enable JavaScript for <code>neverssl.com</code>.
                        </div>
                </div>
        </noscript>
        <div class="header">
                <div class="container">
                <h1>NeverSSL</h1>
                </div>
        </div>
        <div class="content">
        <div class="container">

        <h1 id="status"></h1>
        <script>document.querySelector("#status").textContent = "Connecting ...";</script>
        <noscript>

                <h2>What?</h2>
                <p>This website is for when you try to open Facebook, Google, Amazon, etc
                on a wifi network, and nothing happens. Type "http://neverssl.com"
                into your browser's url bar, and you'll be able to log on.</p>

                <h2>How?</h2>
                <p>neverssl.com will never use SSL (also known as TLS). No
                encryption, no strong authentication, no <a
                href="https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security">HSTS</a>,
                no HTTP/2.0, just plain old unencrypted HTTP and forever stuck in the dark
                ages of internet security.</p>

                <h2>Why?</h2>
                <p>Normally, that's a bad idea. You should always use SSL and secure
                encryption when possible. In fact, it's such a bad idea that most websites
                are now using https by default.</p>

                <p>And that's great, but it also means that if you're relying on
                poorly-behaved wifi networks, it can be hard to get online.  Secure
                browsers and websites using https make it impossible for those wifi
                networks to send you to a login or payment page. Basically, those networks
                can't tap into your connection just like attackers can't. Modern browsers
                are so good that they can remember when a website supports encryption and
                even if you type in the website name, they'll use https.</p>

                <p>And if the network never redirects you to this page, well as you can
                see, you're not missing much.</p>

        <a href="https://twitter.com/neverssl">Follow @neverssl</a>

        </noscript>

        </div>
        </div>

        </body>
</html>
ubuntu@ubuntu-server-1:~$
```

Higher resolution photos can be found in the "images" folder.