# LAB 7

Objective:

1. Get acquainted with some commonly used networking commands and TCP/IP diagnostic tools.
2. Understand the concept of layering/encapsulation by looking at Link, IP and TCP headers.
3. Understand the concept of multiplexing using Ethernet "frame type" field, IP "protocol field", transport "port number" field.

## Exercise 7.1 Play Time

Play around with tcpdump, wireshark, ping, arp, route, ifconfig, host Look at /etc/hostname; /etc/hosts; /etc/network/interfaces; /etc/resolv.conf; /etc/protocols; /etc/services and understand what the files are for.

```
130905628@oslab-23:~$ cat /etc/hostname;
oslab-23
130905628@oslab-23:~$ cat /etc/hosts;
127.0.0.1       localhost
127.0.1.1       oslab-23

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
130905628@oslab-23:~$ cat /etc/network/interfaces;
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback
130905628@oslab-23:~$ cat /etc/resolv.conf;
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
search mahe.manipal.net
130905628@oslab-23:~$ cat /etc/resolv.conf;
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
search mahe.manipal.net
130905628@oslab-23:~$ cat /etc/protocols;
# Internet (IP) protocols
#
# Updated from http://www.iana.org/assignments/protocol-numbers and other
# sources.
# New protocols will be added on request if they have been officially
# assigned by IANA and are not historical.
# If you need a huge list of used numbers please install the nmap package.

ip      0       IP              # internet protocol, pseudo protocol number
hopopt0         HOPOPT                  # IPv6 Hop-by-Hop Option [RFC1883]
icmp    1       ICMP            # internet control message protocol
igmp    2       IGMP            # Internet Group Management
ggp     3       GGP             # gateway-gateway protocol
ipencap         4       IP-ENCAP        # IP encapsulated in IP (officially ``IP'')
st      5       ST              # ST datagram mode
tcp     6       TCP             # transmission control protocol
egp     8       EGP             # exterior gateway protocol
igp     9       IGP             # any private interior gateway (Cisco)
pup     12      PUP             # PARC universal packet protocol
```

```
udp     17      UDP             # user datagram protocol
hmp     20      HMP             # host monitoring protocol
xns-idp         22      XNS-IDP                 # Xerox NS IDP
rdp     27      RDP             # "reliable datagram" protocol
iso-tp4 29      ISO-TP4                 # ISO Transport Protocol class 4 [RFC905]
dccp    33      DCCP            # Datagram Congestion Control Prot. [RFC4340]
xtp     36      XTP             # Xpress Transfer Protocol
ddp     37      DDP             # Datagram Delivery Protocol
idpr-cmtp 38    IDPR-CMTP       # IDPR Control Message Transport
ipv6    41      IPv6            # Internet Protocol, version 6
ipv6-route 43   IPv6-Route       # Routing Header for IPv6
ipv6-frag 44    IPv6-Frag       # Fragment Header for IPv6
idrp    45      IDRP            # Inter-Domain Routing Protocol
rsvp    46      RSVP            # Reservation Protocol
gre     47      GRE             # General Routing Encapsulation
esp     50      IPSEC-ESP       # Encap Security Payload [RFC2406]
ah      51      IPSEC-AH        # Authentication Header [RFC2402]
skip    57      SKIP            # SKIP
ipv6-icmp 58    IPv6-ICMP       # ICMP for IPv6
ipv6-nonxt 59   IPv6-NoNxt      # No Next Header for IPv6
ipv6-opts 60    IPv6-Opts       # Destination Options for IPv6
rspf    73      RSPF CPHB       # Radio Shortest Path First (officially CPHB)
vmtp    81      VMTP            # Versatile Message Transport
eigrp   88      EIGRP           # Enhanced Interior Routing Protocol (Cisco)
ospf    89      OSPFIGP                 # Open Shortest Path First IGP
ax.25   93      AX.25           # AX.25 frames
ipip    94      IPIP            # IP-within-IP Encapsulation Protocol
etherip         97      ETHERIP                 # Ethernet-within-IP Encapsulation [RFC3378]
encap   98      ENCAP           # Yet Another IP encapsulation [RFC1241]
#       99                      # any private encryption scheme
pim     103     PIM             # Protocol Independent Multicast
ipcomp          108     IPCOMP                  # IP Payload Compression Protocol
vrrp    112     VRRP            # Virtual Router Redundancy Protocol [RFC5798]
l2tp    115     L2TP            # Layer Two Tunneling Protocol [RFC2661]
isis    124     ISIS            # IS-IS over IPv4
sctp    132     SCTP            # Stream Control Transmission Protocol
fc      133     FC              # Fibre Channel
mobility-header 135 Mobility-Header # Mobility Support for IPv6 [RFC3775]
udplite 136     UDPLite                 # UDP-Lite [RFC3828]
mpls-in-ip 137          MPLS-in-IP      # MPLS-in-IP [RFC4023]
manet 138                       # MANET Protocols [RFC5498]
hip     139     HIP             # Host Identity Protocol
shim6   140     Shim6           # Shim6 Protocol [RFC5533]
wesp    141     WESP            # Wrapped Encapsulating Security Payload
rohc    142     ROHC            # Robust Header Compression

cat /etc/services >> text1.txt

[file:\\text1.txt]

# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, officially ports have two entries
# even if the protocol doesn't support UDP operations.
#
# Updated from http://www.iana.org/assignments/port-numbers and other
# sources like http://www.freebsd.org/cgi/cvsweb.cgi/src/etc/services .
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.
```

```
tcpmux          1/tcp                   # TCP port service multiplexer
echo        7/tcp
echo        7/udp
discard         9/tcp       sink null
discard         9/udp       sink null
systat      11/tcp      users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd        17/tcp      quote
msp         18/tcp                  # message send protocol

....
```

At the end of this exercise, you should have some basic understanding of how a host manages network information as well as gain some experience on using networking tools. You shouldbe able to collect a trace (write to a file) via tcpdump and view the trace in wireshark (using the -r option).

Exercise 7.2 Simple Stuf

1. Whats your machine's host name and IP address? How did you get this information?
2. What is the next hop router's IP address and MAC address? How did you get this information?
3. What is the local DNS server's host name and IP address? How did you get this informa - tion?
4. What do the numbers in the file /etc/protocols represent?
5. What is the port number associated with applications: ssh, ftp, nfs, smtp (email)?

Exercise 7.3 Encapsulation and Demultiplexing

Goal: To understand layering and demultiplexing, Ms. Lux wants to capture packets. She also wants to understand how web flows operate at the same time. So, help her design an experiment that captures only those packets that are exchanged between her machine and CSE web server when she clicks the url http://mycse.mahe.manipal.net
Guidance:

1. Run tcpdump with -n option to avoid name lookup.
2. Use wget (command: wget --no-proxy http://mycse) to download the url. You could also use firefox/chrome, but this is cleaner and simpler.
3. Your trace should not capture any background traffic.
4. Before answering the questions, explore different packets by clicking on the individual packets. Also note the sequence of packet exchange.

LAB REPORT

1. Explain your experimental design by specifying the exact commands (with options) you will run and in which order. Avoid description unless absolutely necessary.

A. Command: tcpdump -i any -w exercise3.pcap -n '((src host felicity.netne.net) or (dst host felicity.netne.net))'
then: wireshark -r exercise4.pcap

Explanation: First flag is `-i`. This helps us define the interface. We are using `any` so that it gets packets from any interface.
Second flag is `-w` this writes the dump in a file. Then `-n` avoids name lookup and `src host <url>` ensures the packets are sourced from the specified <url>.

2. Select the first TCP packet listed.

a) Which next-hop node is it destined to? Specify the next-hop node's MAC and IP address. How did you determine this information?

A. Next-hop node: Cisco_ed:66:c1 (the router, most probably);
MAC Address: 00:19:56:ed:66:c1, wireshark (next (not final) destination MAC address);
IP Address: 10.105.11.21, the default gateway.

b) Who is the packet's final destination? Specify the final destination's IP address. How did you determine this information? Can you find it's MAC address?

A. The packet's final destination is synerg.cse.iitb.ac.in. Its IP address is 10.129.41.2 (this came from wireshark's first packet's destination attribute). MAC address cannot be determined since it's the first packet (sent) we're analysing.

c) What are the fields used at the link(Ethernet), IP and TCP headers to demux the packet at the next hop or destination? Specify the values of these fields in decimal format and the corresponding process(protocol) the packet is passed to.

A. Ethernet header: Field - IP, value = 2048 (in decimal)
IP header: Field - TCP, value = 6
TCP header: Field - HTTP, value = 80;

3. Apart from the above reporting, name your trace file as "exercise7_1.out" and upload the file to portal.

## Exercise 7.4 More Demultiplexing

Goal: With the success of the previous experiment, Ms. Rani now wants to capture and examine different types of traffic, basically arp, ICMP (protocol used by ping) and ssh. She wants to capture all of the above in just one single trace. Help her design an experiment to do the same.

Guidance:
1. For ssh, you could ssh to your neighbor's machine.
2. In wireshark, click on the protocol field to order the packets according to the protocol.

## LAB REPORT

1. Explain your design by specifying the exact commands (with options) you will run and in which order. Avoid description unless absolutely necessary.

A. tcpdump -i em1 -w exercise4.pcap -n 'arp' or port 22 or 'icmp'
wireshark -r exercise4.pcap

2. Arp protocol: Click on any one of the ARP packets.

a) Trace the flow of this packet up the protocol stack i.e specify what all processes/protocols handle this packet.

A. It starts on the Physical Layer, which is the reason for the Ethernet header. Demultiplexing it gives the ARP field value. Demultiplexing it gives out the IP field, which means it ends up in the IP stack (2048).

b) What is the value of the field used in Ethernet header to pass packets to the ARP module? Express it in decimal format.

A. Hardware length: 6
Protocol length: 4
Opcode reply(2)
sender MAC and IP address followed by target's.

3. ICMP protocol: Click on any one of the ICMP packets.

a) Trace the flow of this packet up the protocol stack i.e specify what all processes handle this packet.

A. Starts off with the ethernet, with the field IP (2048). It then gets passes to IP with the field ICMP (1). Then gets passed on to ICMP removing it's header.

b) Expand the “Ethernet” header. Which higher level process (protocol) is this packet passed to and what is the value in decimals?

A. It's passed on to the Internet Protocol stack and it's value in decimals is 2048.

c) Expand the IP header. What is the value of the field used in this header to pass packets to the ICMP module? Express it in decimal format.

A. Value is 1.

4. SSH protocol: Click on any one of the SSH packets.

a) Click on the IP header field. Specify the source and destination IP addresses.

A. Source: 10.105.1.11; Destination: 10.105.11.21

b) Expand the TCP header. Specify the source and destination port numbers.

A. Source Port: SSH (22)
Destination Port: 39478 (39478)

c) Which machine (IP address) is the SSH server? Hint: SSH server's listen on designated ports as specified in /etc/services.

A. Machine: 10.105.1.11, i.e the machine ssh-ed into.

5. Name your trace file as “exercise4.pcap” and add the file to your roll number directory.