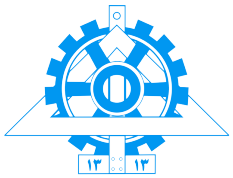
In the name of Allah

بسم الله الرحمن الرحیم

Linux and TCP/IP networking Laboratory Manual



University of Tehran
دانشگاه تهران

School of Electrical and Computer Engineering
دانشکده مهندسی برق و کامپیوتر

Computer Network Lab
آزمایشگاه شبکه‌های کامپیوتری

Dr. Ahmad Khonsari - احمد خونساری
a_khonsari@ut.ac.ir

Amir Haji Ali Khamseh'i - امیر حاجی علی خمسه‌ء
khamse@ut.ac.ir

Muhammad Borhani - محمد برهانی
Amirahmad Khordadi - امیر احمد خردادی
Sina Kashi pazha - سینا کاشی پزها
Mohammad Ali Shahsavand - محمد علی شاهسوند

March 11, 2019

۲۰ اسفند ۱۳۹۷

Systems configuration

Launch GNS3 and make a network as below. You can use "**ifconfig** `eth0` **192.168.0.1 netmask 255.255.255.0**" to set ip.

Table 0.1: The IP addresses of the hosts

Host	IP Address	Subnet Mask
h0 (shakti)	128.238.66.100	255.255.255.0
h1 (vayu)	128.238.66.101	255.255.255.0
h2 (agni)	128.238.66.102	255.255.255.0
h3 (apah)	128.238.66.103	255.255.255.0
h4 (yachi)	128.238.66.104	255.255.255.0
h5 (fenchi)	128.238.66.105	255.255.255.0
h6 (kenchi)	128.238.66.106	255.255.255.0
h7 (guchi)	128.238.66.107	255.255.255.0

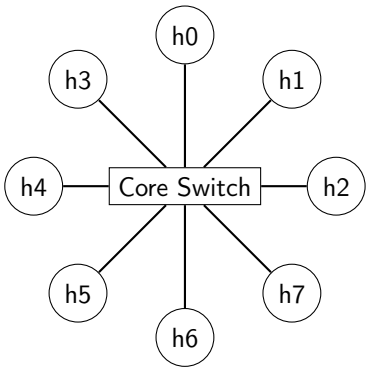


Figure 0.1: A single segment network

1 Telnet service

Run **ps -e** to list the processes running in `h1`. After starting a new process by running **telnet** in another command window, execute **ps -e** again in a third window to see if there is any change in its output.
Find the process id of the **telnet** process you started, by:

ps -e | grep telnet

Then use **kill** *process-id-of-telnet* to terminate the **telnet** process.

Report

What is Internet service daemon (inetd)?
Is **inetd** started in your system? Why?
Is **xinetd** started in your system? What is its PID?

2 Default network services

Display the file `/etc/services` on `h1` screen, using:

more `/etc/services`

Then in another console, use the redirect operator to redirect the **more** output to a file using **more** `/etc/services > ser-more`. Compare the file `ser-more` with the original **more** output in the other command window.
Copy `/etc/services` file to a local file named `ser-cp` in your working directory, using **cp** `/etc/services ser-cp`. Compare files `ser-more` and `ser-cp`, using **cmp** `ser-more ser-cp`. Are these two files identical?
Concatenate these two files using **cat** `ser-more ser-cp > ser-cat`.
Display the file sizes using **ls -l ser***. Save the output. What are the sizes of files `ser-more`, `ser-cp`, and `ser-cat`?

3 Network command manual

Read the **man** pages for the following programs:

1. arp
2. arping
3. ifconfig
4. tcpdump
5. ping
6. netstat
7. route
8. wireshark
9. iptables

Study the different options associated with each command. Throughout this lab you will use these commands rather extensively.

Report

Explain the above commands briefly. Two or three sentences per command would be adequate.

4 Packet capturing

In this exercise, we will use **tcpdump** to capture a packet containing the link, IP, and TCP headers and use ethereal to analyze this packet.
First, run **tcpdump -enx -w** `dump.out` in `h1`. You will not see any **tcpdump** output, since the **-w** option is used to write the output to the `dump.out` file.
Then, you may want to run **telnet** `10.0.0.2` to generate some TCP traffic.[1] After you login to `h2`, terminate the **telnet** session and terminate the **tcpdump** program. Next, you will use **wireshark** to open the packet trace captured by **tcpdump** and analyze the captured packets. To do this, run **wireshark** `dump.out &`. The **wireshark** Graphical User Interface (GUI) will pop up and the packets captured by **tcpdump** will be displayed. Select any one of the packets that contain the link, IP, and TCP headers.

Report

What is the value of the `protocol` field in the IP header of the packet you saved? What is the use of the `protocol` field?
What is the value of the `frame type` field in an Ethernet frame carrying an IP datagram?

5 ARPing

This time we will run wireshark to capture an ARP request and an ARP reply in real-time. Simply run **wireshark &** in `h1` and select the interface and start capturing. If there is no arp requests and replies in the network, generate some using **arping** *10.0.0.2*.
Now you should see several ARP replies in the arping output.

Report

What is the value of the `frame type` field in an Ethernet frame carrying an ARP request and in an Ethernet frame carrying an ARP reply, respectively?
What is the use of the `frame type` field?

6 Packet filtering

Using the **tcpdump** utility, capture any packet on the LAN and see the output format for different command-line options. Study the various expressions for selecting which packets to be dumped.
For this experiment, use the **man** page for **tcpdump** to find out the options and expressions that can be used. If there is no traffic on the network, you may generate traffic with some applications (e.g. **telnet**, **ping**, etc.).

[1] Remember to run `/etc/init.d/xinetd` **restart** in `h2` to start telnet server on it.

Report

Explain briefly the purposes of the following **tcpdump** expressions. If using wireshark, use next list

- **tcpdump udp port 520**
- **tcpdump -x -s 120 ip proto 89**
- **tcpdump -x -s 70 host** `ip_addr1` **and (**`ip_addr2` **or** `ip_addr3`**)**
- **tcpdump -x -s 70 host** `ip_addr1` **and not** `ip_addr2`

If you ar using **wireshark** explain the following filter.

- **udp.port == 520**
- **ip.proto == 89**
- **ip.addr ==** `ip_addr1` **and (ip.addr ==** `ip_addr2` **or ip.addr ==** `ip_addr3`**)**
- **ip.addr ==** `ip_addr1` **and not ip.addr** `ip_addr2`

7 Connection port

In `h1` run **wireshark &** and select an interface to capture packets between hosts.
Execute a TCP utility, **telnet** for example, in another command window:
telnet *10.0.0.2*

Report

What are the port numbers used by the `h1` (local machine) and `h2` (remote machine)?
Which machine's port number matches the port number listed for **telnet** in the `/etc/services` file?

8 Random port

In `h1` run **wireshark &** and select an interface to capture packets between hosts.
Then, **telnet** to the `h2` from a second command window by typing **telnet** *10.0.0.2*. Again issue the same **telnet** *10.0.0.2* command from a third command window. Now you are opening two **telnet** sessions to `h2` simultaneously, from two different command windows.
Check the port numbers being used on both sides of the two connections from the output in the **wireshark** window.

Report

When you have two **telnet** sessions with your machine, what port number is used on the `h2` (remote machine)?
Are both sessions connected to the same port number on the `h2` (remote machine)?
What port numbers are used in `h1` (local machine) for the first and second **telnet**, respectively?
Explain briefly what a `socket` is.