Exercises: Basics of Networking Tools

Experiential Learning Workshop

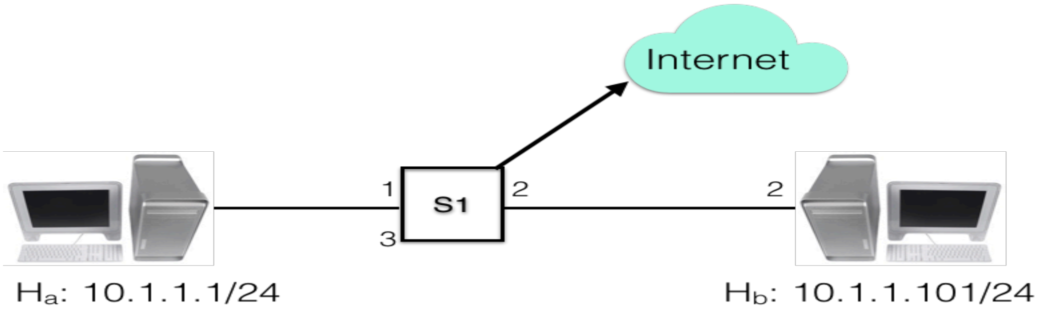
1 General Guidelines

1. Make a team of two or three unless stated otherwise.
2. **For** each exercise, use wireshark capture to verify contents
3. **Ensure** to use proper capture filter and don't capture irrelevant traffic
4. **Where** appropriate or applicable, use `wget` or `nc` to access the web server.
5. **The** default client for accessing web server is assumed to be browser, preferably firefox. You can use Chrome or any other browse as well.
6. **The** webserver in the example below is taken as 'myweb.com'. Please use your hostname or corresponding IP address instead in your exercise.
7. **To** kill any program in the linux terminal, please press **`Ctrl-C`** and not **`Ctrl-Z`**. The latter will suspend the program and not stop it.

Note: Appendix provides instructions on installing any package if not already installed.

2 Hands-on 1: Tools

The exercises below assume that client's IP address is 10.1.1.1 and other machine in your team has the IP address 10.1.1.101. Please use appropriate IP address in your setup.



2.1 Understand use of Wireshark to analyze network traffic

2.1.1 General usage and invocation.

1. Open wireshark (you might have to open in **sudo** mode)
2. Select the applicable interface e.g. `enp0s1` or `eth0`.
3. Specify the capture filter to capture traffic with other e.g. '`host 10.1.1.101`' if the IP address of other host is 10.1.1.101.
4. Click start
5. Access a web page from this website on the browser. For example, enter http://10.1.1.101.

2.1.2 Accessing websites

1. Browse your institute's website, e.g. `www.ieee.org` and capture traffic for this website. Use the appropriate capture filter (host http://www.ieee.org/). You should see traffic for this website only and no other traffic. Analyze the capture and look at how many packets are exchanged.

2. Access other websites e.g. vtu.ac.in or and other websites you prefer. Define appropriate capture filter for the same. Wireshark should show only relevant packets and not all kind of traffic.

2.1.3 Use other capture filters

1. Use a filter to exclude traffic from a web site e.g. host not 10.1.1.101 and analyze captured traffic.

2.1.4 Using other options

1. Use display filter to see traffic for a tcp stream.
2. Save some packets into a file and reopen the file
3. Explore various options of display time format.
4. Explore options of ordering packets by different fields e.g. by src address, by INFO field, by packet length etc.

2.1.5 Exercise Expectations

1. Launch wireshark and select the active interface.
2. Able to specify the appropriate capture filters and thus capture packets only of interest
3. Able to sort the packets w.r.t. different fields e.g. info, src, dst, time, etc.
4. Able to save packets to a file and open the file for later analysis.
5. Able specify display filter and see packets of interest.

2.2 Using ping

Note: Wherever count (-c option) is not specified, use Ctrl-C to abort.

1. Always Use wireshark to analyze all the traffic for below steps.
2. Ping `google.com` and `yahoo.com` by sending some fixed count packets e.g. 20. Analyze the response times and variation in response times.
3. Ping these sites again in quite mode (option -q). Analyze the packet loss.
4. Use ping with changing interval duration to 0.2s from the default of 1s as well as changing packet size from 56bytes to 1000 bytes.
5. Use flooding option (-f) to ping local m/c on the network. Analyze in wireshark the options of time difference between packets.
6. Use ping to send a packet of different size e.g. 1000 bytes with our own data pattern. Analyze the response time.
7. Use ping to use a different source IP address (e.g. of your neighbor) and analyze the response.

2.2.1 Exercise Expectations

1. Able to use ping with its various options.
2. Able to analyze ping response variation and packet loss statistics.

2.3 Using nc

1. Open terminal on two machines.
2. Identify each other's IP address. You can use the command `ip addr` in the linux terminal, to know the IP address of Ethernet interface. Do not the IP 127.0.0.1 for `lo` interface.

3. Run as TCP server on some port e.g. 2345 (`nc -l 2345`) in one terminal and UDP server (`nc -u -l 3456`) in another terminal.
4. Connect using clients (from another machine) to both TCP and UDP server and do chat.
5. Analyze wireshark capture of your chat conversation.
6. Transfer some files across machines e.g. `cat` "file"| `nc` "server IP" "server Port" on the client side and on server side (`nc -l` "port" >"file")
7. Login in to remote machine without authentication

2.3.1 Exercise Expectations

1. Able to use nc for both TCP and UDP.
2. Able to do file transfer between two machines in the quickest possible way instead of using pendrive.
3. Able to use to communicate with a web server.

2.4 Using wget

1. Open terminal (command line. Preset Ctrl-Alt-T or from menu)
2. Mimic (option `-mk`) your college website (e.g. http://www.ksit.ac.in/), and access locally (turn off your internet).
3. Download a large file using the `--limit-rate=1m` e.g. http://rprustagi.com/workshops/web/media/movie.mp4, break the download by pressing **Ctrl-C** after about 5MB is downloaded and then download with resume option (-c). Ensure full download occurs and see if you can watch the movie after complete download.
4. Explore other options such as –d for debug headers, -O to save into a file,

2.4.1 Exercise Expectations

1. Able to use **wget** to download contents of a website for offline use.
2. Able to resume broken download.

← end of exercise handout →