#### COEN 146: Computer Networks

**Lab 2: Network Commands, Packet delays, and HTTP**

**Objectives**

##### To demonstrate network commands

##### To measure packet delays

##### To setup client-side connection to HTTP server

##### 

##### **Guidelines**

##### For all COEN 146L you need to develop a good knowledge of C in a Linux development environment. You are highly encouraged to use command line tools in developing, compiling, and running your programs.

##### Please pay attention to your coding style and good programming practices, if your program is not worth documenting, it will not worth running.

##### **Network commands and tools**

1. [20%] Run each of the following basic networking commands in Linux, and write down your observation and explain the usage of the command
   1. netstat: displays the contents of network interfaces, with – a option, the command displays the state of all active sockets (more to come on sockets as end points of communication). With – r option, the routing table is displayed. Please type man netstat to learn about all options.
   2. ifconfig: configures network interface parameters. With – a option, the state of all interfaces are displayed. Other options are also used to assign a new IP address to an interface, to assign a new network mask for an interface, to disable an interface, and more. Please type man ifconfig to learn about all options.
   3. hostname: displays and sets hostname of the system. Please type man hostname to learn about all options.
   4. ping: sends ECHO\_REQUEST datagram to a network host using ICMP protocol to elicit an ECHO\_RESPONSE from the host or gateway. Please type man ping to learn about all options.
   5. traceroute: displays the route packets trace to a network host using IP protocol “time to live”. Please type man tracerout to learn about all options.
   6. telnet: remote connection to server at a specific port (mainly 80 – http port)
   7. host/dig: performs DNS lookups.
   8. route: manipulates network routing tables.
   9. arp: displays and modifies the Internet-to-Ethernet address translation tables used by the address resolution protocol.

Note: some of the commands may require that you are logged in admin priviliges.

1. [20%] Select three hosts in the internet (one in North America, one in Asia, and one in Europe), and experiment with pinging each host with different packet sizes ranging from 32-bytes to 1048-bytes. For each host,
   1. identify the packet loss.
   2. identify the RTT “Round-Trip-Time”.
   3. Explain the correlation between these measurements to the geographical location of the hosts.
2. [20%] Suppose that the IP address for one of the URLs you selected in Step 2 is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. Suppose that three DNS servers are visited before your host receives the IP address from DNS. The first DNS server visited is the local DNS cache, with an RTT delay of RTT0 = 3 msecs. The second and third DNS servers contacted have RTTs of 20 and 26 msecs, respectively. Initially, let's assume that the Web page associated with the link contains exactly one object, consisting of a small amount of HTML text. Suppose the RTT between the local host and the Web server containing the object is RTTHTTP = 47 msecs.

Write a C program that computes the following:

* 1. Assuming zero transmission time for the HTML object, how much time elapses from when the client clicks on the link until the client receives the object?
  2. Now suppose the HTML object references 6 very small objects on the same web server. Neglecting transmission times, how much time elapses from when the client clicks on the link until the base object and all 6 additional objects are received from web server at the client, assuming non-persistent HTTP and no parallel TCP connections?
  3. Assume that the client is configured to support n parallel TCP connections, compute the response time in both persistent and non-persistent cases.

1. [20%] Setup a client side to connect to **gaia.cs.umass.edu 80**, as follows:
   1. Type **telnet gaia.cs.umass.edu 80** to connect to gaia.cs.umass.edu web server**,** explain what happens

Note: If you are using MacOS terminal, you may need to install telnet binary file to /usr/local/bin/

* 1. Type a GET HTTP request:

**GET /kurose\_ross/interactive/index.php HTTP/1.1**

**Host: gaia.cs.umass.edu**

Write down your observation.

* 1. Analyze the response message sent by the HTTP server, by answering the following questions:

 What is the name of the file that is being retrieved in this GET message?

 What version of HTTP is the client running?

 What formats of text and images, if any?

1. [20%] Setup clients to connect to the three hosts you selected in Step 3, then experiment to connect to the public http port 80 and other ports, say: 3389. Write down your observations.

**Requirements to complete the lab**

1. Demo to the TA the commands, http connection, and correct execution of your programs [recall: a successful demo is 25% of the grade]
2. Submit the source code of your program as .c file and your analysis and observations in a .txt file and upload to Camino.

Please start each program/ text with a descriptive block that includes minimally the following information:

# Name: <your name>

# Date: <date> (the day you have lab)

# Title: Lab1 – task

# Description: This program computes … <you should

# complete an appropriate description here.>