**PART I**

------------------------------------------------------------------------------------------------------------------------------------------

**Ping:**

Command: ping <domain name>

Average times:

www.yahoo.com: 18ms

www.calvin.edu: <1ms

Given that the lab computers are connected to the same local network as the servers for Calvin’s website, it takes less time to send packets to and from the server.

Command: ping /?

Flags:

-t: “Ping the specified host until stopped.”

-a: “Resolve addresses to hostnames.”

-n #: “Number of echo requests to send.”

I used the -t flag and had to use CTRL+C to get it to stop. I let it run until the whole command line window had been filled with reports of the

**Tracert:**

Command: tracert www.yahoo.com

Example result:

7 6 ms 6 ms 6 ms be1.core-01.dtw.ussignalcom.net [74.204.127.254]

This command showed that sending packets to www.yahoo.com took 16 steps on their way. I never realized how many servers it takes for information to get from one device to another over the internet. Without thinking about it, I always assumed that information went from one computer to some central server and then directly to the desired server or computer. It makes sense in hindsight that it takes more than one or two servers to make internet connections.

**Nslookup:**

Command: Nslookup <domain name>

Result:

www.calvin.edu:

Server: ulu.calvin.edu

Address: 153.106.4.99

Non-authoritative answer:

Name: www.calvin.edu

Address: 153.106.4.23

**Whois:**

Results:

|  |  |
| --- | --- |
| **Registrant Org** | Calvin College |
| **Dates** | 10,766 days old Created on 1989-04-25 Expires on 2019-07-31 Updated on 2018-09-26 |
| **IP Address** | 153.106.4.23 - 1 other site is hosted on this server |

**Hostname:**

Command: hostname

Result: maroon27

As an extra test for this one, I ran this command: ping -a 127.0.0.1 and got

“Pinging maroon27.csad.cs.calvin.edu …”

**IPConfig:**

Command: ipconfig /all

This command allows the user to see the IP address or addresses for their computer. When I ran the command, it showed the device IP address as well as the default gateway IP (ss64.com/nt/ipconfig.html).

Example Result:

IPv4 Address…….: 153.106.116.237(Preferred)

Default Gateway………: 153.106.116.1

**Netstat:**

Command: netstat -na

“Display current TCP/IP network connections and protocol statistics” (ss64.com/nt/netstat.html).

The -n flag displays the addresses and ports as numbers instead of as domain names

The -a flag displays all of the connections and listening ports available.

Example result:

TCP 192.168.56.1:16451 192.168.56.1:1855 ESTABLISHED

**Arp:**

Command: arp -a

“Display and modify the IP-to-Physical address translation tables used by address resolution protocol” (ss64.com/nt/arp.html).

The -a flag displays all the ARP entries (same reference). The entries are displayed in an organized table.

**Managing Domain Names:**

ICANN stands for the Internet Corporation for Assigned Names and Numbers, which is the organization that accredits domain name registrars.

I used www.networksolutions.com and found that www.ryanholwerda.com is an available domain name.

------------------------------------------------------------------------------------------------------------------------------------------

**PART II**

------------------------------------------------------------------------------------------------------------------------------------------

The server client is listening to port 11000 for a connection from the client.

The GetHostName() method gets the hostname and IP address from both the server and the client and allows them both to know which machine they’re connecting to.

The connection uses port 11000.

The client sends the GET<EOF>:# and the server sends the client the message that the data was received, at which point the client adds to the counter and sends again.

When the server receives the STOP<EOF>:10 message, it knows to stop listening to port 11000 and closes the connection.

The server and client communicated and ran a total of 10 times.

**Separate workstations:**

Server Hostname: maroon28

The server did not respond, so I altered the server computer’s firewall to allow incoming connections through port 11000. Once the new firewall rule was in place, the client and server were able to communicate normally.