**Internet Exploration (15 Points)**

**IS175**

**Goal:** This document will walk you through the use of command line utilities and through C# code that will demonstrate the use of sockets. You will need to provide responses to questions and notes about observations. You will create a 500 word (or so) document in Microsoft Word that will describe your observations, questions and general thoughts about working through this document. Place your name at the top of the document. Note: Please make sure what you prepare is neat, proof read and with a **professional look**. Please use single spacing with an 11 point font size.

**Part I – Command Line Utilities (10 Points)**

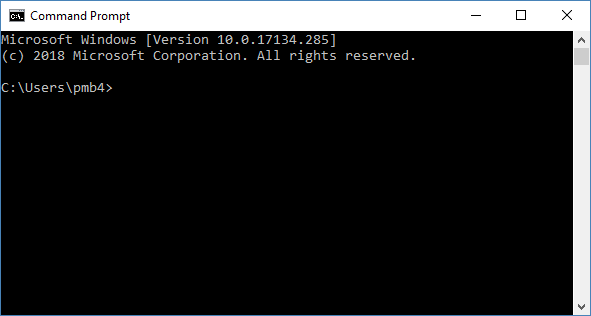
This exercise will explore the use of some command line and utilities that support communication and information collection on the internet. Apply the lecture discussion, take time to conduct research through Google for more information about the commands, and note what you discover.

To start, place your mouse cursor in the command search box:



Then enter **cmd** to start a command window.

After that, you will see an icon at the top showing you the name of a program called “cmd.exe” Select that, and you will see a display similar to the following:



As you are told to enter commands below, key them in as asked. Again, note your observations, what you may have found out through some light research, responses to some questions in this document and thoughts about what the command is doing, its value as a utility etc. There is no right or wrong answer. Just take time to use your curiosity. ☺

**PING**: The ping program checks to see if another host on the network is reachable. Sometimes it is called the “echo” utility. In its simplest form, ping sends packets to another host. If the other host is accepting ping packets (and if it is available to the network), the other host will send back verification that it received the packets. The ping command is also useful informing you of how long it may take to connect to another host, and it is especially useful as the first tool to diagnose connectivity. Try the following commands one at a time and pay attention to the amount of time it takes:

ping [www.yahoo.com](http://www.yahoo.com)

ping [www.calvin.edu](http://www.calvin.edu)

Comment on the difference between the times. Why do you think there is a difference?

Now enter the command below. This will provide a brief (admittedly terse) description of more ways to use the ping command.

ping /?

Observe some of the options and try to use one. Make sure you place a space between the command, its “flags” and the computer name you are trying to reach. Note which one you used and comment.

**TRACERT:** This command (short for trace route) lets you look at how many places your packets had to stop at to make a connection to another host on the internet. Try entering the following command line:

TRACERT www.yahoo.com

**Information about websites**: There are a number of tools available to gain information about another host. One tool is **nslookup** which will provide you the numeric IP address. Try using it in the two examples:

nslookup [www.yahoo.com](http://www.yahoo.com)

nslookup [www.calvin.edu](http://www.calvin.edu)

In each of these you may notice the first entry for a “Server”. That is the server that is providing the address lookup service (remember domain name server (DNS)). The other addresses are the known addresses for the host referenced.

Another handy command is the **whois** command. Generally, it’s not available on MS Windows, but is available through other websites. So, if you’re getting emails from an unknown source, you can use **whois** to get information about the server the emails originated from. One website that provides an easy to use interface is <http://whois.domaintools.com/> Go to that website and try entering the following URL’s:

[www.](http://www.xinyout.org)ripparoo.com

[www.calvin.edu](http://www.calvin.edu)

[www.sqe.com](http://www.sqe.com)

**Your Workstation’s Name:** Usually a PC on a network has been given a name. To find out that name, enter **hostname.** Note the name of your PC.

**What’s Happening on Your Machine:** Some commands provide you information about your own PC and the connections currently made. Try entering the following commands:

ipconfig /all

netstat –na

arp -a

After reviewing the output of the commands, take time to research (google) the commands. Discuss what the commands are intended to do, and what are you noting about your own observation of the commands.

**Managing Domain Names:** Read the following Wikipedia article up to the Table of Contents: <http://en.wikipedia.org/wiki/Domain_name> You will see a link for domain name registrars. Follow that link. Read the first paragraph and also research the rest of that article to determine what is ICANN (note what you find out in your paper). Then go to the list of accredited registrars at <http://www.icann.org/registrar-reports/accredited-list.html> . As you scroll down, you will see a list of DNS suffixes. Scroll further and you will see a list of companies that provided domain name registration. Look for any organization that supports domain names for the United States, click on their link. Now, try to determine if a domain name based on your name is available (e.g. for Patrick Bailey, patrickbailey.com or patbailey.com would be searched). Was it available? How many variations did you have to try to find a domain name for yourself?

**Part 2 – Review C# Code for a Socket**

It is recommended you review the following links:

* <https://docs.microsoft.com/en-us/dotnet/framework/network-programming/sockets>
* <https://docs.microsoft.com/en-us/dotnet/framework/network-programming/how-to-create-a-socket>
* <https://docs.microsoft.com/en-us/dotnet/framework/network-programming/using-client-sockets>

On Moodle, you will a zip file named **CSCode.** Download it to a local folder and **Extract All** of its contents. You will see two folders: ConsoleServer and ConsoleClient.Open up both solutions. Briefly ConsoleServer is an application that opens a socket to **Listen** for a client application to connect to it. Take time to examine the code. ConsoleClient is the complement. It is looking for a Server to connect to. Again, take time to look at the code. Here are some things to consider:

* What port number is the Server opening?
* How does the Server and the Client know the name of the host?
* What is the port number being used?
* What message does the client send to the server? How many times does it send that message?
* How does the Server know when to stop “talking” to the client and go back to listening?

Now run the ConsoleServer with **Start without Debugging.** Then do the same for ConsoleClient.Observe the interaction. Stop both applications. Verify they are “talking” to each other.

Now let’s run the Console and Server on separate work stations. First, change the following line in Program.cs file in ConsoleServer solution:

IPAddress ipAddress = ipHostInfo.AddressList[0];

To

IPAddress ipAddress = ipHostInfo.AddressList[1];

Then, determine the domain name of the machine where you will run ConsoleServer.

Next, on the workstation where ConsoleClient is installed change the ConsoleClient file of Program.cs . Assuming the workstation where ConsoleServer is running is name maroon34, you would change Program.cs in the ConsoleClient solution as follows:

Change

IPHostEntry ipHostInfo = Dns.GetHostEntry(Dns.GetHostName());

To

PHostEntry ipHostInfo = Dns.GetHostEntry("maroon34");

Try running the applications as before, only this time the client and server are on separate workstations. Most likely, you might get a message that the server is not responding. This is because our workstations have firewalls. To address the problem, read the article at <https://www.online-tech-tips.com/windows-10/adjust-windows-10-firewall-settings/> You will need to change the firewall settings on the machine running the server. While you can’t turn off the Firewall, you can create a **Rule** for a workaround. So, in that link focus on the section labeled **Advanced Firewall Settings**. Note in your document what changes you made to get the two applications talking to each other.

Now submit your document to Moodle. Also, reboot your machine if you are doing this in our lab.