**INFS 3338 HOMEWORKS – Spring 2017**

**All homeworks should have this instruction page as a cover sheet, with each item checked off as you complete it. Each printout should be correctly labeled for the step it corresponds to from the instructions.**

**Section 3338 02i (remote ITV section) can omit the cover sheet, but MAKE SURE you still correctly label each step according to the numbers in the instructions.**

**H1 - Address identification**

1. **On a Windows PC, open a DOS command window, and run 'ipconfig /all'**
2. **Take a screenshot displaying the full visible output results**
3. **PRINT YOUR SCREENSHOT**
4. **Highlight, circle or mark, and label each of**
	1. **MAC address**
	2. **IPv4 address**
	3. **Default gateway**
5. **Repeat steps 1-4 above on two more, different computers. One of your reports can be the same three pieces of information on your smartphone. Collect, print and label the required addresses**

**Notice that this homework requires a total of three separate, correctly labeled printouts.**

**DUE January 26**

**H2 - Packet Capture**

1. **Open Wireshark, browser, DOS command window**
2. **Use Google to find the ICAAN whois website. Find the registration info for utrgv.edu.**
3. **Copy ALL the registration info and copy/paste it into a MS Word document.**
4. **Start Wireshark capture on your active interface. Browse www.utrgv.edu website a little bit to get some packets.**
5. **Stop the capture after a minute or two. Sort by the Protocol column.**
6. **Find an HTTP protocol packet with the UTRGV IP address showing, sending a JPEG image. Identify the application layer payload where it indicates a JPEG. Take a screenshot showing both the top pane with the packet line, AND the middle pane showing the packet layers.**
7. **PRINT YOUR SCREENSHOT. Mark and label the UTRGV address and the JPEG payload.**
8. **Repeat steps 6 and 7 for a PNG (portable network graphics) packet.**
9. **Find a TCP protocol packet with the UTRGV IP address showing, sending an ACK message.**
10. **PRINT YOUR SCREENSHOT. Mark and label the UTRGV address and the ACK message.**

**DUE January 26**

**HW3 – Network information in Windows**

1. **Logon to a computer running Windows, preferably one where you have admin privileges. Make sure your machine is currently connected to a network (can be a wired LAN or wifi, either one).**
2. **Go to Control Panel. In ‘Classic View’ (all the icons shown individually), select the Networking icon. Find your ‘Local Area Connection’. This looks slightly different in different versions of Windows, but they all use ‘Local Area Connection’ as the label or link. Click on the link or the ‘view status’ link, depending.**
3. **You will get a popup called ‘Local Area Connection Status’. SCREENSHOT AND PRINT.**
4. **This page will show the network you are currently connected to, this might be ‘domain network’ for a school, home or workplace wired network; or, it might be ‘public network’ if you are connected to a wi-fi hotspot. Clearly mark and label the network you are connected to. For example, my office PC is connected to ‘ds.utpa.edu’. Find this information on your printout and clearly mark it as your currently connected network.**
5. **Click on the ‘Details’ button. You will get a second popup box called ‘Network Connection Details’. SCREENSHOT AND PRINT.**
6. **On your printout, clearly mark and label your IPv4 address, your default gateway, and your MAC address.**
7. **Go back to the first Control Panel screen and find Device Manager. This can be reached through the ‘System’ icon, upper left corner. Open the Device Manager.**
8. **Go down the list to find your ‘Network Adapters’. Click on the ‘+’ sign to open up the information. Depending on how your PC is configured, you likely have several adapters. If you are currently connected using wi-fi, find your wi-fi adapter and select it. SCREENSHOT AND PRINT, and label it as ‘my wifi adapter’. Check with DOS window ipconfig /all if you’re not sure.

If you’re currently using a wired connection, find your Ethernet adapter and select it. SCREENSHOT AND PRINT, and label it as ‘my Ethernet adapter’. Again, check with DOS window ipconfig /all to if you’re not sure about the maker of your device. One of my machines uses Intel, another has Realtek; there are other possible makers.**

**Due January 31**

**H4 - FTP Client - Server Connection**

1. **For this exercise, you may use either a Linux terminal (e.g. Ubuntu) or the Filezilla client.**

 **NOTE: On my laptop, I was only able to ‘get’ a file when either the application client was started**

 **with ‘Run As Administrator’. You can try it without, but start as admin if you run into problems**

 **such as a ‘permission denied’ message.**

1. **Find a public ftp server – typically ‘anonymous’ logon.**

**NOTE: In Filezilla, the default display in the top pane does NOT show the server response codes. Fix this by right-clicking in the top pane, and select ‘Show detailed log’. The output will now include additional lines (green in my version) with response codes such as 220, 250, 226, etc, with accompanying explanatory text.**

1. **Submit a request to the server to connect - SCREENSHOT showing IP and port. PRINT and MARK the paper to identify and SEPARATELY LABEL the IP address and port.**
2. **Login successfully - SCREENSHOT showing you're connected. PRINT and MARK and LABEL the paper to show the server response that indicates successful connection.**
3. **Navigate to a directory containing a file you can download. SCREENSHOT and PRINT, and MARK and LABEL the paper to identify some of the files made available for download on the site.**
4. **Download a file. SCREENSHOT and PRINT, and MARK and LABEL your work showing the server response indicating the successful file transfer.**
5. **Navigate to your local directory where the downloaded file is now located. SCREENSHOT and PRINT, and MARK and LABEL the paper to identify the file you obtained.**

**DUE Feb 2**

**H5 - Netcat Web Server**

 **1. Open Ubuntu. This can be a bootable flash drive, or a VMWare virtual machine, or a dual-**

 **boot PC on which you have already installed Ubuntu. Make sure your NIC is in Bridged mode**

 **if you're using VMWare.**

 **2. If you haven't already, install 'netcat':**

 **sudo apt-get install netcat**

**3. Start the 'gedit' program with Admin rights:**

 **sudo gedit**

**4. Create a simple web page. Type in this code as it appears here. Where it says (your name) below**

 **write your own name:**

**<html>**

 **<head>**

 **<title>INFS 3338 NETCAT DEMO</title>**

 **</head>**

 **<body>**

 **<h1>This is a Netcat webpage</h1>**

 **<h2> (your name)</h2>**

 **</body>**

**</html>**

**5. Save this file with the filename 'index.html'.**

**6. Use the 'cat' command to display the file index.html**

 **cat index.html**

 **SCREENSHOT AND PRINT**

**7. Create a netcat web server on port 80, using index.html as 'Standard In' input to the server:**

 **sudo netcat -l -v 80 < index.html**

 **The terminal will show a response that it is listening.**

 **SCREENSHOT AND PRINT - HIGHLIGHT and LABEL the line which reports that it's listening**

**8. Open a second terminal window (don't close the first terminal). Run a network status to show**

 **that the local server is listening on port 80:**

 **netstat -tln**

 **SCREENSHOT AND PRINT - HIGHLIGHT and LABEL the line showing the open port**

**9. Run ifconfig to learn your current IP address.**

 **SCREENSHOT AND PRINT - HIGHLIGHT and LABEL the line showing your IP address.**

**10. Open Firefox browser in Ubuntu. Type the IP address from step 9 directly into the**

 **URL line and hit enter. Your netcat webpage will display.**

 **SCREENSHOT AND PRINT. HIGHLIGHT and LABEL netcat server IP address in the URL line.**

**In case it doesn't work the first time, you'll need to restart the netcat web server, using the same**

**command in step 7. This will need to be done each time you want to load the page again. You can**

**use the !! command to repeat the immediately previous command.**

**DUE Feb. 2**

**H6 – Making a TCP Connection**

1. **For this exercise you need a Linux Ubuntu platform with admin rights. Use your bootable flash drive, VMWare, or a lab machine. If in VMWare, make sure you’re using NAT: Player – Manage – Virtual Machine Settings – Network Adapter.**
2. **If you haven’t already, install netcat:**

**sudo apt-get install netcat**

1. **Use netcat as a client to send a TCP packet to google.com on port 80:**

 **(Always use the “verbose” switch, -v)
netcat –v google.com 80**

1. **SCREEN CAPTURE AND PRINT, and MARK AND LABEL the line in your terminal showing that your connection has succeeded.**
2. **Open an additional terminal, and set them up side by side so that you can capture both at once. Run nslookup on google.com and identify the IP address you are connecting to.**

 **SCREEN CAPTURE AND PRINT, and MARK AND LABEL the line showing the IP address.**

1. **The next step requires timing, so have your commands ready to go. Google will only leave you connected for a short time, so you need to have your follow-up ready.**

**Left Terminal you will run: netcat –v google.com 80
Right Terminal, type in but don’t hit Enter yet: netstat –tan**

1. **Execute the netcat first, then immediately the netstat. Your new connection will be visible in the list of connections. Scroll to where you find the google IP address connected to your own IP.**

**SCREEN CAPTURE AND PRINT, and MARK AND LABEL the line showing the IP addresses. Correctly mark and identify YOUR address, and separately mark and label GOOGLE’s addr.**

1. **Run your netcat command again to make a connection to Google. When the cursor drops to the next line, type in (2 separate lines)**

**HEAD / HTTP/1.1**

**host: www.google.com**

**Then hit return TWICE.**

 **You will see several lines of output showing the response of the web server. This is the “header”**

 **information of the packet, without the actual HTML content. One of the lines will begin with**

 **“Server:”, and the server listed for google.com will be “gws”. This is Google’s own custom web**

 **server designed for their own use.**

1. **Now run through this again three more times, for –**

**www.baruch.cuny.edu 80, host is www.baruch.cuny.edu**

**www.utb.edu 80, host is www.utdallas.edu**

**www.rice.edu 80, host is www.rice.edu**

1. **For EACH site, do another SCREEN CAPTURE AND PRINT, and CLEARLY MARK AND LABEL the line which shows what server is running the web response for that site. (3 more printouts, total).**

**DUE Feb 9**

**H7 Using socketAPI – Sending data from a Client Socket with Python to a Netcat Server**

1. **Open Ubuntu (VMWare, bootable flash drive, dual-boot etc), and a terminal.**
2. **First, make a netcat listener on localhost port 33333. Include the commands to send whatever it receives on this port to a file called mySocketMessage using the ‘standard output’ method:**

 **netcat –v –l localhost 33333 > mySocketMessage**

1. **Open a new, separate terminal, and check that your listener is working, using netstat:**

**netstat –tpan**

**SCREEN CAPTURE AND PRINT**

**MARK AND LABEL YOUR OUTPUT – identify host, port, and program that is running**

**If you need to start over, you may need to kill a process still running on the port # you tried. Remember that you can find PID# with netstat –tpan, and you can kill a process with**

**kill –KILL (PID # goes next, no parentheses)**

1. **Next, start Python by typing ‘python’ at the command prompt (without the quotes, of course). You will see the Python triple prompt: >>>**

**Now Python is ready to execute commands as you enter them one by one**

1. **In order to have access to the socketAPI code, we need to import the socket binaries:**

**>>> import socket (then hit Enter)**

1. **As you did before in H6, create a socket object using the socket class, socket method of that class, and the two arguments which tell it to use standard IP addresses and TCP for connections:**

**>>> mySocket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)**

1. **Create a variable which stores the location we will be connecting \*\*to\*\*:**

**>>> myPort = (‘localhost’, 33333)**

1. **This time, instead of binding to a port (something you do with a server), we will be connecting to the netcat Listener we already set up. For this we need the socketAPI ‘connect’ command:**

**>>> mySocket.connect(myPort)**

**SCREEN CAPTURE AND PRINT**

**SHOW \*\*BOTH\*\* TERMINALS AT ONCE**

**MARK AND LABEL THE LINE SHOWING YOUR CONNECTION WAS ACCEPTED**

1. **Now that we’re connected to the Listener, it’s time to send through some data—remember, this data will go directly into the file mySocketMessage from step 2 above.**
2. **To transmit data through the connection, use the socketAPI ‘send’ command:**

**>>> mySocket.send(“This data was transmitted by Python to netcat on port 33333\n”)**

**After you hit ‘Enter’, the Python terminal you’re running will print the number of bytes transmitted.**

**However, nothing appears on the receiving end that has the netcat Listener. The data we sent should now be stored in a file called mySocketMessage. Demonstrate that the transfer was successful by opening a NEW, SEPARATE TERMINAL – THIS IS YOUR 3rd TERMINAL OPEN AT ONE TIME. Use the ‘cat’ Linux command to display the file contents:**

**~$ cat mySocketMessage**

**SCREEN CAPTURE AND PRINT SHOWING YOUR PYTHON AND ‘cat’ TERMINALS**

**MARK AND LABEL the ‘send’ command you just used in the Python terminal**

**MARK AND LABEL the ‘cat’ command and results you just observed in the 3rd terminal**

1. **Send an additional line using the socketAPI ‘send’ command:**

**>>>mySocket.send(“This data is concatenated with the existing file as a second line.\n”)**

**In your 3rd terminal, run ‘cat’ again to see the file contents, with your extra line added.**

**SCREEN CAPTURE AND PRINT SHOWING YOUR PYTHON AND ‘cat’ TERMINALS**

**MARK AND LABEL the ‘send’ command you just used in the Python terminal**

**MARK AND LABEL the ‘cat’ command and results you just observed in the 3rd terminal**

**DUE Feb 14**

**H8 – Understanding Network Services**

**1. Research the ‘qotd’ protocol.**

**2. Find the RFC which provides the specification for the qotd service**

 **PRINT OUT THE SPECIFICATION (all)**

**3. Write a brief report containing a table which provides the following information:**

* **What port is qotd assigned to**
* **What does the service provide (what does it do for a connected client)**
* **How is the service closed (Under what conditions does the server close the connection)**
* **What type of data is sent**
* **What are the constraints on the data sent back (any limitations?)**
* **What syntax is required**
* **What connection protocols are supported**

**Use this format:**

|  |  |
| --- | --- |
| **Question** | **Answer** |
| * **What port is qotd assigned to**
 |  |
| **etc** |  |

**4. Use netcat to create a client TCP connection to cygnus-x.net on port 17 (see previous homeworks for**

 **netcat instructions). As soon as you do you will receive the quote of the day.
 SCREEN CAPTURE AND PRINT**

 **MARK AND LABEL THE COMMAND YOU ENTERED TO CONNECT TO PORT 17**

**DUE Feb 16**

**H9 – Using network tools**

**You will need Linux for this homework. Any distro is okay, as long as you have the commands available which are used below.
You can run Linux as a bootable flash drive, as a dual-boot partition installed on your computer, or as a virtual machine under VMWare.**

**I have a simple webpage at** [**https://faculty.utrgv.edu/j.hughes**](https://faculty.utrgv.edu/j.hughes)

1. **Start your Linux machine and open a terminal.**
2. **Use netcat to see if you can connect with a TCP packet on port 80. Use verbose mode.**

**SCREEN CAPTURE AND PRINT**

 **MARK AND LABEL NETCAT COMMAND.**

**Leave connection open for the next steps**

1. **Open a 2nd terminal and use nslookup to find out the IPv4 address of the web server.**

**SCREEN CAPTURE AND PRINT**

 **MARK AND LABEL THE IP ADDRESS OF THE SERVER.**

1. **Find out your own machine’s current IPv4 address.**

**SCREEN CAPTURE AND PRINT**

 **MARK AND LABEL YOUR IP ADDRESS**

1. **Check your network connections to see your established connection to faculty.utpa.edu on port 80.**

**Use p, n, and t options in netstat. Check to see that the IP address matches.**

**SCREEN CAPTURE AND PRINT.**

 **MARK AND LABEL YOUR IP ADDRESS**

 **MARK AND LABEL IP ADDRESS OF faculty.utpa.edu WEB SERVER**

 **MARK AND LABEL THE PORT YOU’RE CONNECTED WITH ON YOUR PC**

1. **Ctrl-Z in the first terminal to break the connection.**

**Use netcat to create a new connection on port 80 to collect my webpage and pipe the data to a new file called myHTML.txt**

**When the connection is made, enter the command to get my index.htm page:**

**GET /j.hughes/index.htm HTTP/1.1**

**host: faculty.utrgv.edu**

**(return)**

**(return)**

**Notice that this is htm, not html. When you hit Enter, netcat will get the HTML code, put it into the myHTML.txt file, and close. You will now see the standard prompt.**

1. **Demonstrate that the file content was collect and saved by using the cat command.**

**SCREEN CAPTURE AND PRINT**

**There will be too much content for one screen – Just scroll so that mostly HTML is showing, and screenshot that. The first HTML line should begin with <!DOCTYPE html PUBLIC (etc)**

1. **In your terminal window, click and drag to highlight ALL/ONLY the HTML content, from <!DOCTYPE at the top to </html> at the very end. Right-click and select ‘Copy’**

**Open another terminal and type ‘gedit’ to start the text editor program.
Put the cursor in the gedit workpane and right-click, select ‘Paste’.**

**You should now see all the HTML code again.**

**Find the spot near the top of the <body> section where it has my name in all caps:**

**JERALD HUGHES**

**Replace this with your own name. You may make other changes if you like, but be careful not to ‘break’ the HTML with incorrect code. NOTE: The banner image will not appear, unless you download that with the correct name referenced in the HTML, and place in the same folder as the HTML; this can be done by other means, such as a right-click while the real page from the web server appears in the browser.**

**Put your cursor in the top of the gedit window so that the menus appear.**

**Select File – Save As – and save it as myHTML.htm**

1. **Open Firefox in Ubuntu. Put your cursor at the very top of the Ubuntu screen where it shows “Firefox Web Browser”, to open the menu line. Select File – Open. Find your file and open it.**

**It should now be showing my webpage, but with your name in it.**

**SCREEN CAPTURE AND PRINT.**

**MARK AND LABEL THE CHANGES YOU MADE – You don’t have to screenshot every single part of the page, just get most of it, and make sure your changes are visible.**

 **Due Feb 21**

**HW 10 - Web server interaction**

**Repeat steps 6 through 9 from HW9 above, this time using this webpage as a target:**

[**http://www.linuxcommand.org/index.php**](http://www.linuxcommand.org/index.php)

**Reports:
1. PRINT, MARK AND LABEL a screenshot of your terminal showing your netcat connection**

 **command, with the pipe.**

**2. Demonstrate your successful capture with the cat command, like step 7 above. SCREENSHOT**

 **AND PRINT as in step 7.**

**3. gedit your HTML content, as in step 8 above. Put your name in it here or there. Include a**

 **reference to UTRGV, another to INFS 3338. The precise changes don’t matter; the point is to**

 **play with editing the HTML. SCREENSHOT, MARK AND LABEL YOUR CHANGES IN**

 **HTML SOURCE.**

 **NOTE: Again, the images will not load in your customized version, unless you download them**

 **and save them with correct names in the same folder as your version.**

**4. Save as an .htm file, open in Firefox. MARK AND LABEL YOUR CHANGES.**

**Due Feb 21**

**HW11 – FTP commands – Passive Mode for Temporary Data Connection Port**

**You will need Linux for this homework. Any distro is okay, as long as you have the commands available which are used below.
You can run Linux as a bootable flash drive, as a dual-boot partition installed on your computer, or as a virtual machine under VMWare.**

1. **Start your Linux system (virtual machine is fine) and open two terminals.**
2. **Use netcat to connect to ftp.utexas.edu on port 21. SCREENSHOT, MARK AND LABEL to show you’re connected.**
3. **Use the FTP commands USER and PASS to logon as ‘anonymous’ with no password.**
4. **Use PWD to display your current directory. Use CWD to navigate to /pub/freebsd/. SCREENSHOT, MARK AND LABEL to show you have reached the target folder.**
5. **In the second terminal, prepare to connect to the FTP server for its listing for the freebsd folder. You need the IP address for ftp.utexas.edu (use nslookup to get this number), and you won’t know the port number yet. Don’t run netcat yet, because you still need that number.**
6. **In the first terminal, send the PASV command. The FTP server will respond with a list of 6 integers. Use integers 5 and 6 to determine the port (int#5 \* 256, plus int#6 = target port#).**
7. **In the second terminal, add the required port number, and \*now\* run the netcat. SCREENSHOT, MARK AND LABEL showing your PASV command and the result from the server, plus your netcat in the second terminal with the results of your calculation for the target port.**
8. **In the first terminal, request the directory listing for the freebsd folder by using the command**

**LIST –a**

**SCREENSHOT, MARK AND LABEL showing the directory listings in the second terminal.**

1. **One of the files there is README.TXT**
2. **Repeat the PASV process to get ready to collect this file and store it on your computer. This time, not only must you calculate the port # for the second terminal, you also have to pipe the results to your own file (use “>”). On the first terminal therefore, you will be using first the PASV command again.**
3. **Once you have the numbers for the port calculated and entered, your second terminal will connect, ready to receive the data and pipe it; then you run the RETR command to get your own copy of the file. The netcat tool should pipe the file to your machine using the name ‘freeBSDmessage.txt’. Execute the command. SCREENSHOT, MARK AND LABEL again showing the response to your PASV command in the first terminal, and your full command to hit the right target port and pipe the result to a file in the second terminal.**
4. **You should now have freeBSDmessage.txt in your home directory. Use the cat command in Linux to display this text in the second terminal. It’s long file, so don’t try to print the entire thing. Scroll back up in the terminal so that your netcat command and the first dozen or so lines of the file are visible. It starts with “Welcome to the freeBSD archive!” SCREENSHOT, MARK AND LABEL showing your netcat command, and the output of the cat command.**

**Due April 6, 2017**

**HW12 – Email ports**

**You will need Linux for this homework. Any distro is okay, as long as you have the commands available which are used below.
You can run Linux as a bootable flash drive, as a dual-boot partition installed on your computer, or as a virtual machine under VMWare.**

**MARK AND LABEL THE IP addresses AND the ports separately, individually, for each netstat:**

 **your IP, your port3, remote IP, remote port#**

**1. Find out what URL America Online uses for its SMTP server.**

**2. Use nslookup to get the IPv4 address.**

**3. Use netcat with the IPv4 address to make a simple TCP connection to the standard (old) SMTP port.**

 **Remember to specify the assigned port for SMTP, not the port for web or other services.**

 **SCREEN CAPTURE AND PRINT – MARK AND LABEL**

**4. In a second terminal, use netstat –pnt to check the established connection to that port.**

 **Remember, you need to do netstat fairly quickly, before the connection is dropped.**

 **SCREEN CAPTURE AND PRINT - MARK AND LABEL**

**5. Find out what URL America Online uses for its POP server. Get the IPv4 with nslookup.**

**6. Use netcat with the IPv4 address to make a simple TCP connection to the standard POP port.**

 **Again, remember to specify the right number for the assigned port for the POP service.**

 **SCREEN CAPTURE AND PRINT – MARK AND LABEL**

**7. In a second terminal, use netstat –pnt to check the established connection to that port.**

 **SCREEN CAPTURE AND PRINT - MARK AND LABEL**

**8. Use netcat with the IPv4 address to make a simple TCP connection to the secure POP port.**

 **SCREEN CAPTURE AND PRINT – MARK AND LABEL**

**9. In a second terminal, use netstat –pnt to check the established connection to that port.**

 **SCREEN CAPTURE AND PRINT - MARK AND LABEL**

**10. Find out what URL America Online uses for its IMAP server. Get the IPv4 with nslookup.**

**11. Use netcat with the IPv4 address to make a simple TCP connection to the standard IMAP port.**

 **SCREEN CAPTURE AND PRINT – MARK AND LABEL**

**12. In a second terminal, use netstat –pnt to check the established connection to that port.**

 **SCREEN CAPTURE AND PRINT - MARK AND LABEL**

**13. Use netcat with the IPv4 address to make a simple TCP connection to the secure IMAP port.**

 **SCREEN CAPTURE AND PRINT – MARK AND LABEL**

**14. In a second terminal, use netstat –pnt to check the established connection to that port.**

 **SCREEN CAPTURE AND PRINT - MARK AND LABEL**

**DUE April 6, 2017**

**HW13 Understanding DNS Due April 13, 2017**

**1. On a Linux system (can be a virtual machine), open a terminal. Type 'nslookup' to start the nslookup program. Type 'set type=ANY' to collect all available DNS records. Change the DNS server you will be querying by typing 'server 8.8.8.8', which is Google's public DNS server.**

**2. type 'rit.edu' to get all the information from Rochester Institute of Technology. PRINT SCREENSHOT. FIND, MARK AND LABEL:
The address AND port number of the DNS server which provided the information.**

 **The IPv4 address of rit.edu**

 **The IPv6 address of rit.edu**

 **A nameserver of rit.edu**

 **A mail exchange server of rit.edu**

**3. Choose one of the nameserver addresses, and change to that server. Request the IP address for utrgv.edu, in order to demonstrate that this domain nameserver, which is AUTHORITATIVE for its domain, has no information about utrgv.edu. SCREENSHOT PRINT. FIND, MARK AND LABEL**

 **The IP and port # of the server which provided the response.**

 **The response which indicates that no information is provided.**

**4. Still using the rit.edu nameserver, request information about saunders.rit.edu, to indicate that this AUTHORITATIVE server does indeed have information about this URL, owned by RIT. SCREENSHOT AND PRINT. FIND, MARK AND LABEL**

 **The IP address and port # of the server which provided the information.**

 **The IPv4 address of saunders.rit.edu.**

 **Type 'exit' to close the nslookup program.**

**5. Type 'host -a utrgv.edu' to get all the information about utrgv.edu. SCREENSHOT AND PRINT.**

 **FIND, MARK AND LABEL**

 **The server and port # which provided the information. (hint: look at the end of the output, use your knowledge of the DNS port number)
The IPv4 address for utrgv.edu**

 **A nameserver for utrgv.edu**

 **A mail exchange server for utrgv.edu**

**6. Type 'dig ANY @8.8.8.8 tamu.edu' to use the Google public DNS server to get all the info on A&M.**

 **SCREENSHOT AND PRINT. FIND, MARK AND LABEL:**

 **The server and port # which supplied the information (again, look toward the end of the output, remembering the port number for DNS)**

 **The IPv4 address for tamu.edu (this will be an ‘A’ type address)**

 **A nameserver for tamu.edu (‘NS’ type)**

 **A mail exchange server for tamu.edu (‘MX’) type**

**7. Type 'dig' with no arguments, to see the list of root server URL's. Find the IPv4 address of the 'a' root**

 **nameserver. SCREENSHOT, MARK AND LABEL**

**8. Use the 'dig' command with 'AAAA' as an additional option, to find the IPv6 address of the 'a' root**

 **nameserver. SCREENSHOT, MARK AND LABEL**

**HW 14 - Understanding Ethernet Frames**

1. **Open a browser, DOS command window, and Wireshark while connected to Internet. Use the university network eduroam, since this generates a lot of network traffic for a variety of protocols.**
2. **Run ipconfig /all, PRINT, and MARK AND LABEL your IPv4 address**
3. **Start Wireshark capture on your connected interface**
4. **Generate some traffic by going to NetworkWorld, ping google.com. Give it a couple of minutes, so that you at least have ARP, HTTP and ICMP packets. Stop the capture.**
5. **Filter Wireshark to only a 'ping' packet by entering "ICMP" in the filter field. We will be using the middle and lower panes, so adjust size as necessary to make sure the data can be seen in your printouts. Select one ICMP packet.**
6. **The middle pane may have layers expanded or not depending on what you were doing the last time you used Wireshark. For our purposes, you want the Frame, Internet Protocol, and ICMP layers (first, 3rd, 4th) collapsed, showing only one line and a '+' next to them. Fully expand the 2nd layer, the 'Ethernet' layer, so that the Destination and Source lines are also fully expanded.**
7. **In the expanded Ethernet (in layer model, the 'Network' layer) info, PRINT, find and CLEARLY MARK AND LABEL each of the following, BOTH middle and lower panes:**
	1. **the Ethernet (Network Layer) information**
	2. **the Internet Protocol (Internet layer) information**
	3. **the ICMP payload information**
8. **Clear the filter, apply an 'arp' filter. Arrange the layers the same way: Ethernet layer fully expanded, other layers fully collapsed.**
9. **In the expanded Ethernet info, PRINT, find and CLEARLY MARK AND LABEL each of the following, BOTH middle and lower panes:**
	1. **6 bytes of destination address**
	2. **6 bytes of source address**
	3. **2 bytes of type code . Look up this code and write in what this code stands for**
	4. **padding bytes: How many does it need? Write this in.**
10. **Clear the filter, apply an 'http' filter. Arrange the layers the same way: Ethernet layer fully expanded, other layers fully collapsed. Use the 'Info' column in the top pane to find a 'GET' packet. Click on that packet line in the top pane.**
11. **In the expanded Ethernet info, PRINT, find and CLEARLY MARK AND LABEL each of the following, BOTH middle and lower panes:**
	1. **6 bytes of destination address**
	2. **6 bytes of source address**
	3. **2 bytes of type code . Look up this code and write in what this code stands for**
12. **Same packet, PRINT again, then CLEARLY MARK AND LABEL each major section in both the middle and lower panes, with the number of bytes present in each.**

**How many bytes are there in:**

* 1. **the entire frame**
	2. **the Ethernet II (Network) layer**
	3. **the Internet layer**
	4. **the TCP (Transport) layer**
	5. **the Hypertext Transfer (Application) layer**

**Write your numbers in the appropriate spots on your printout.**

**Make sure your b,c,d,e add up to a.**

**HW15 ARP information Due April 20, 2017**

1. **For this homework, you need to be connected to a network - easiest choice is some wi-fi, such as school or home.**
2. **Open a DOS window as administrator. Run ipconfig /all. SCREENSHOT and PRINT, MARK AND LABEL both your current IP address and MAC address.**
3. **Run the command:**

**arp -a**

**...to see the current entries in your ARP table. SCREENSHOT AND PRINT. MARK and LABEL the "Interface:" entry which shows the IP address of your currently connected NIC.**

**MARK and LABEL the IP address of your default gateway**

**MARK and LABEL the MAC address of your default gateway**

1. **Run the command:**

**net view**

**...to see the names of other machines on the network.
In DOS output, the machine names are preceded by a double backslash: \\**

**Ping one or two of these machines using the machine name, WITHOUT the backslash.**

**SCREENSHOT AND PRINT, MARK and LABEL the IP address of a machine which responded to the ping.**

1. **For the next step you need to be quick, because your ARP table will repopulate quickly.
Delete all the entries in ARP table by entering the command:**

**netsh interface ip delete arpcache**

**AS SOON as the command completes with "Ok", IMMEDIATELY run the command**

**arp -a
If you're fast enough, you'll see "No ARP entries found". SCREENSHOT AND PRINT.
---If you can't catch it fast enough for zero entries, but it just has one entry (usually your gateway), this will be acceptable.**

1. **Ping your gateway, DNS server, DHCP server (these are probably the same at home), and a couple of more machines you either found in the previous ARP run, or from the net view command. This will repopulate your ARP cache.**
2. **Run the command**

**arp -a**

**one more time. SCREENSHOT AND PRINT, MARK and LABEL the IP and MAC addresses of at least one other machine on the network.**