ECE 560 Homework 1

Guanhua Chen

TOTAL POINTS

98 / 100

QUESTION 1

1 Question 1: Internet Standards 2.5 / 3

- **0 pts** Correct

✓ **- 0.5 pts** **Needed some elaboration. No example for NIST**

- **1 pts** Needed more elaboration

QUESTION 2

2 Question 2: A Model for Computer Security 7 / 7

✓ **- 0 pts** **Correct**

- **0.5 pts** One question not answer correctly or fully
- **1 pts** One incorrect answer

QUESTION 3

3 Question 3: Threats and Attacks 12 / 12

✓ **- 0 pts** **Correct**

QUESTION 4

4 Question 4: IP Addressing 5.5 / 6

- **0 pts** Click here to replace this description.
- **0.5 pts** Why Duke uses NAT needed more explanation

✓ **- 0.5 pts** **Why Duke uses NAT explanation is incorrect**

QUESTION 5

5 Question 5: Physical Addresses 5 / 5

✓ **- 0 pts** **Correct**

QUESTION 6

6 Question 6: Networking Protocols 8 / 8

✓ **- 0 pts** **Correct**

- **0.5 pts** Incorrect networking tool (program) that uses ICMP
- **1 pts** No illustrations for TCP setup/teardown
- **0.5 pts** Never mentions that TCP/UDP are transport layer protocols
- **0 pts** TCP = TRANSMISSION control protocol
- **1 pts** No TCP/UDP headers
- **0.5 pts** Need more detail on ARP
- **1 pts** No teardown for TCP handshake
- **0.5 pts** No ICMP protocol header
- **0.25 pts** Does not address important differences between TCP and UDP
- **0.5 pts** Insufficient detail on ICMP
- **0.25 pts** Doesn't talk about teardown in TCP handshake
- **0.75 pts** Lacking most relevant details on TCP/UDP

QUESTION 7

7 Question 7: Ports 8 / 8

✓ **- 0 pts** **Correct**

- **1 pts** Port example?
- **0.25 pts** Off by one error (port numbers)
- **1 pts** Wrong organization (IANA)
- **1 pts** Missing crucial part of port definition (application level multiplexing)
- **0.5 pts** No port number range specified

QUESTION 8

8 Question 8: DNS 8 / 8

✓ **- 0 pts** **Correct**

- **1 pts** Looking for programs that query DNS, not pieces of the network

QUESTION 9

9 Question 9: Network Traffic Analysis with Wireshark 3 / 4

- **0 pts** Correct

✓ **- 1 pts** **Incorrect HTTP request (Should see obvious**

info from course website, and the IP for the course site is 152.3.72.105)

- 1 pts The second snapshot should show HTTP protocol details, not TCP

- 1 pts Incorrect HTTPS request (Should be same dest. IP as HTTP.) (For the course site, 152.3.72.105)

- 0.75 pts Should be able to see source/dest IP for HTTPS, otherwise no way to verify

- 1 pts The third snapshot should show HTTPS requests, not TCP. HTTPS appears to be TLSv1.2 in wireshark, since it can be referred to as "HTTP over TLS".

- 1 pts Incorrect observation. The major difference is that the messages and type of requests are encrypted in HTTPS, which is not in HTTP.

- 0.75 pts Incorrect observation. The major difference is that the messages and type of requests are encrypted in HTTPS comparing with HTTP.

QUESTION 10

10 Question 10: Network Traffic Analysis with TCPDump 3 / 3

✓ **- 0 pts Correct**

QUESTION 11

11 Question 11: Network Mapping 7 / 7

✓ **- 0 pts Correct**

- 0.5 pts Minor Parameter Explanation Error

- 1 pts Major Parameter Explanation Error

- 1 pts No Windows result shown

- 1 pts Minor Error on Explanation to service

- 0.5 pts Scan the wrong site

- 1 pts No Windows Service Explanation

- 0.5 pts Missing Screenshots

- 1 pts No target service explanation

- 3 pts Did not scan Windows VM

- 0 pts Do one scan and explanation correctly, and you get full credit.

https://piazza.com/class/kd6irc9vkbl23b?cid=61

QUESTION 12

12 Question 12: Ncat, Telnet, Netstat, and Sockets 12 / 12

✓ **- 0 pts Correct**

QUESTION 13

13 Question 13: Banner Grabbing: Services Spilling Their Guts 8 / 8

✓ **- 0 pts Correct**

- 1 pts Missing screenshot for the command

- 6 pts No zip file submitted to Sakai

QUESTION 14

14 Question 14: Networking Tools 9 / 9

✓ **- 0 pts Correct**

QUESTION 15

15 Late Penalty 0 / 0

✓ **- 0 pts No penalty**

Question 1: Internet Standards (3 points)

In Chapter 0 and Appendix C of the course textbook, we begin to look at technology standards and standard-setting organizations. Various organizations are involved in the development of standards related to data and computer communications. It is important to understand who the major organizations are and the standards they are responsible for. These standards bodies will be heavily referenced throughout the course and can be useful references when trying to understand different security technologies. **Give a short description of each organization, its key primary responsibilities around standards, and an example of a security-related standard that it has developed**.

a. NIST
   A physical science lab that works with industry to set the standard.
   They work with industry to make sure that the products are measurable. Therefore, they help boost innovation.
   They develop the cybersecurity framework which gives a high-level overview of what a system should do to stay secure. It is designed for server room & board room, meaning that it's kind of abstract idea we should kept in mind.
b. ISOC
   An NGO try to keep the internet a force for good.
   It develops open standard to promote internet. Everyone can provide internet service under the standard.
   It develops time security standard, which helps on accurate timekeeping. It's important because TLS or digital signature needs depend on accurate time.
c. ITU-T
   An organization focus on telecommunication and information communication.
   It mainly focus on communication security.
   X.509 defines framework for PKI & PMI.
d. ISO
   An NGO for global standard.
   It has branches for cyber security.
   ISO/IEC 27032 address roles of different securities in the cyber space.
e. ICANN
   A nonprofit organization maintaining the database for namespace and numerical spaces for the internet.
   Mainly for naming coordination.
   It has a guideline for transparency guidelines which handles vulnerabilities that threaten the security, stability or resiliency.
f. IEEE
   A professional association for EE and allied fields.
   It mainly develops, defines and reviews EE & CE standards.
   It has a cybersecurity community focus on cybersecurity. IEEE Security & Privacy provides practical articles.

1 Question 1: Internet Standards **2.5 / 3**

- **0 pts** Correct

✓ - **0.5 pts** **Needed some elaboration. No example for NIST**

- **1 pts** Needed more elaboration

Question 2: A Model for Computer Security (7 points)

Logwatch is a tool that sends summaries of Linux system logs to an administrator for review. Examine the sshd authentication failures from the Logwatch report below from my home server; this listed reflects a single day's traffic:

```
 ################### Logwatch 7.4.2 (02/27/16) ####################
        Processing Initiated: Tue Aug 14 17:14:03 2018
        Date Range Processed: yesterday
                              ( 2018-Aug-13 )
                              Period is day.
        Detail Level of Output: 0
        Type of Output/Format: stdout / text
        Logfiles for Host: doc
 ##################################################################

--------------------- pam_unix Begin ------------------------

 sshd:
    Authentication Failures:
        root (221.194.47.239): 339 Time(s)
        root (122.226.181.166): 294 Time(s)
        root (115.238.245.8): 258 Time(s)
        root (221.194.44.232): 237 Time(s)
        root (221.194.47.236): 222 Time(s)
        root (115.238.245.4): 212 Time(s)
        root (115.238.245.14): 200 Time(s)
        root (121.18.238.115): 193 Time(s)
        root (112.85.42.196): 192 Time(s)
        root (221.194.44.211): 180 Time(s)
        root (115.238.245.2): 162 Time(s)
        root (112.85.42.201): 144 Time(s)
        root (221.194.47.233): 122 Time(s)
        root (122.226.181.164): 105 Time(s)
        root (122.226.181.165): 90 Time(s)
        root (119.249.54.217): 73 Time(s)
        root (121.18.238.123): 57 Time(s)
        root (122.226.181.167): 54 Time(s)
        unknown (212.83.137.197): 40 Time(s)
        root (221.194.47.221): 39 Time(s)
        unknown (91.121.147.228): 14 Time(s)
        root (212.83.137.197): 10 Time(s)
        unknown (82.99.244.68): 7 Time(s)
        unknown (121.78.144.178): 7 Time(s)
        unknown (188.167.160.166): 6 Time(s)
        unknown (190.202.114.106): 6 Time(s)
(Listing continues for another ~300 lines)
```

Answer the following questions by mapping each of the security concepts in Figure 1.2 from the textbook to the data in the Logwatch report.

1. What is the **asset** we wish to protect?
   The access to the computer facility & access to the file in that server.

2. Who are the **owners** of the asset?
   Prof Tyler.

3. What is the **risk**?
   Unauthorized disclosure. If someone try to break root password through brute force then they have access to the computer.
   Disruption. If someone keep trying to request from the server, it may not be able to serve normal request.

4. What is the **threat**?

   Unauthorized disclosure. Trying to access server as root/unknown.
   Disruption. So many failed login may slow down the respond time from server.

5. What are possible **countermeasures** (prevention, detection, and recovery) to reduce the risk for this threat?
   Configure server so that it only allows ssh-key login.
   Block those IP addresses who fail the authentication for certain amount of times.

6. Using an online IP address locator, for each of the five highlighted entries in the LogWatch report, find what country and country code did each **threat agent** appear to originate from. What Regional Internet Registry are each of the **threat agents** from?
   ```
   221.194.47.239  China CN APNIC
   91.121.147.228 France FR RIPE NCC
   82.99.244.68 IRAN IR RIPE NCC
   188.167.160.166 SLOVAKIA SK RIPE NCC
   190.202.114.106 VENEZUELA VE LACNIC
   ```

2 Question 2: A Model for Computer Security **7 / 7**

✓ **- 0 pts** **Correct**

- 0.5 pts One question not answer correctly or fully

- 1 pts One incorrect answer

Question 3: Threats and Attacks (12 points)

Review the following blog posts by Brian Krebs on https://krebsonsecurity.com/ related to the 2013 Target Data Breach.

- Sources: Target Investigating Data Breach
- Who's Selling Credit Cards from Target?
- A First Look at the Target Intrusion, Malware
- A Closer Look at the Target Malware, Part II
- New Clues in the Target Breach
- Target Hackers Broke in Via HVAC Company
- Email Attack on Vendor Set Up Breach at Target

You may also refer to other articles in the series as needed.

Give a summary of the overall Target data breach including major timelines of the breach.

Nov.15, 2013: Hackers hacked in HVAC company network, which have access to Target internal network

Dec 2, 2013: Logs showed hackers got access to Target POS & an infected server. The malware in affected POS collect credit card info & upload to the infected server. The hacker then used FTP to download data from that server.

From Nov.27-Dec 15, 2013, hackers kept collecting data. Then Target realized the problem.

Referring to Section 1.2 of the textbook, describe the threat consequence(s) and type of threat action(s) that caused the consequence(s) for the data breach outlined.

Unauthorized disclosure: Intrusion. The hacker first broke into HVAC company network.
Deception: Masquerade. The hacker then accessed the Target internal network as if he was actual user from the HVAC.
Unauthorized disclosure: Interception. Then he stole sensitive data from Target.

3 Question 3: Threats and Attacks 12 / 12

✓ **- 0 pts** **Correct**

Question 4: IP Addressing (6 points)

1. What is an IP address?
   ID for the device connected to Internet.
   You need to know sender & receiver when you try to communicate. For example, if I try to communicate with Prof Tyler, at least I need to know his name & he knows my name. Then we both can acquire the info we need. IP serves similarly.
   IP addresses are assigned by network administrators when connected to internet.

2. Using the command line, determine the public IP address of your VM. Include a screenshot.

```
gc171@vcm-16036:~$ dig +short myip.opendns.com @resolver1.opendns.com
67.159.94.111
```

3. What are the two common versions of IP protocols? Show the header for each.
   IPv4. Source: Wikipedia. Version is constant 4

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP						ECN		Total Length															
4	32	Identification																Flags				Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															
24	192																																
28	224																																
32	256																																

   IPv6 source: Wikipedia. Version is constant 6

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	*Version*				*Traffic Class*								*Flow Label*																			
4	32	*Payload Length*																*Next Header*								*Hop Limit*							
8	64	*Source Address*																															
12	96																																
16	128																																
20	160																																
24	192	*Destination Address*																															
28	224																																
32	256																																
36	288																																

4. How many bits and bytes are in IPv4 and IPv6 addresses? How many possible IP addresses are in IPv4 and IPv6?
   ipv4: 32bits 4bytes 2^{32}
   ipv6: 128bits 16bytes 2^{128}

5. IP addresses are divided into 5 category classes, which is called classful addressing. What are the 5 different classes of IP addresses and their ranges?

Class	Address range	Supports
Class A	1.0.0.1 to 126.255.255.254	Supports 16 million hosts on each of 127 networks.
Class B	128.1.0.1 to 191.255.255.254	Supports 65,000 hosts on each of 16,000 networks.
Class C	192.0.1.1 to 223.255.254.254	Supports 254 hosts on each of 2 million networks.
Class D	224.0.0.0 to 239.255.255.255	Reserved for multicast groups.
Class E	240.0.0.0 to 254.255.255.254	Reserved for future use, or research and development purposes.

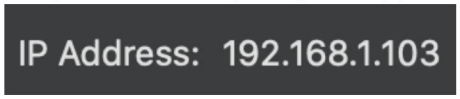
Source: https://www.computerhope.com/jargon/i/ip.htm

6. What is a private IP address? What are the 3 private IP address ranges?
The addresses used in private network.

192.168.0.0 - 192.168.255.255 (65,536 IP addresses)
172.16.0.0 - 172.31.255.255 (1,048,576 IP addresses)
10.0.0.0 - 10.255.255.255 (16,777,216 IP addresses)

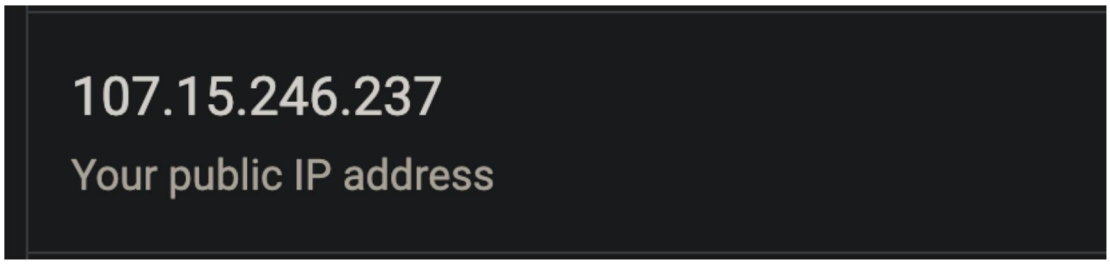
Source: Google

7. Most Duke wifi is in a private IP address pool. Using the command line on your personal computer, determine your IP address (include a screenshot). What private IP address range is it in? Why do you suppose that range was chosen for this environment?



It is in 192.168.0.0-192.168.255.255
I think there are limited devices at home, so the router just uses the smallest range.

8. What is the IP address of the router serving your personal computer? Show a screenshot of how you determined this.



Google. This should be my router WAN ip.

9. Explain what NAT is and why it is important in the context of IPv4 addressing.
   network layer translation. i.e. ip address space remapping.
   It's mainly for scalability & management. Imagine you need to replace your upstream router, then you need to re-assign the ip for the whole subnet without NAT. It is costly. However, with NAT, you just need to change the ip in the packet header. It is much affordable. Also you can assign the ip within the border, a great way to accommodate more device.

10. Does Duke use NAT? What is your evidence that they do or do not?
    Yes.
    For such a large scale system, it makes no sense without NAT. There is basically no downtime and that is impossible without NAT. Re-assigning ip for whole network will take some time.
    Also with so many devices running in the campus, you need NAT to better scale up ip namespace.
    Finally it can act as firewall. The duke network open limited port to outside. E.g. you cannot visit windows vcm (RDP tcp 3389) outside campus.

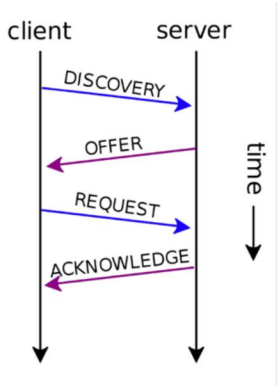
11. Some examples of special IP address groups are: Multicast, Loopback Address, and Link Local. What are they and their range(s)?
    multicast: Distributing content to multi-clients by router/switch
    224.0.0.0 – 239.255.255.255

    loopback: routing packets back to source
    127.0.0.1

    Link local: the ip for communication within the same network segment
    169.254.0.0-169.254.255.255

12. There are two common ways for a computer to get an IP address: it may be set statically on the computer, or it may request one from the network. What is the latter approach called and how does it work?
    DHCP.

Overview: When client connected to routers/vpn server, they will assigned the device with unique ip address.
Details:



Clients send ip broadcast “DISCOVERY”(destination 255.255.255.255 udp port 67)
DHCP server send “OFFER” with ip config
Clients send a “BROADCAST” request to one offer
DHCP send “ACKNOWLEDGE”

Now clients has ip address and basic config
Source: prof Tyler slides

4 Question 4: IP Addressing 5.5 / 6

- **0 pts** Click here to replace this description.
- **0.5 pts** Why Duke uses NAT needed more explanation

✓ **- 0.5 pts** **Why Duke uses NAT explanation is incorrect**

Question 5: Physical Addresses (5 points)

1. Explain what a MAC Address is.
   The physical unique address for every device.

2. What are MAC Addresses for your Linux VM? For your personal computer?
   linux vm: 00:50:56:a1:87:25
   My pc: 82:36:0b:25:10:01

3. How many bits and bytes are in a MAC Address?
   48bits 6bytes

4. What is significant about the first three bytes of a MAC Address?
   It represents the manufacturer

5. Using the first three bytes of this MAC address of your Linux VM's eth0 interface, give the manufacturer of this NIC (Network Interface Card) as given by the IEEE OUI. We already know it's a VM, but what hypervisor product is hosting the VM?
   Vmware,Inc

5 Question 5: Physical Addresses 5 / 5

✓ - **0 pts** **Correct**

Question 6: Networking Protocols (8 points)

1. What is ICMP and what is the common networking tool that uses this protocol? Show the ICMP protocol header.
   A protocol to generate the error message to source ip when network problems prevent ip packets delivery.
   Tool: ping, traceroute

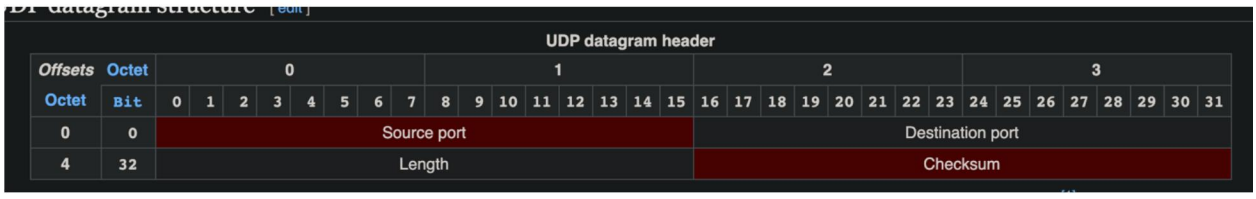
Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Type								Code								Checksum															
4	32	Rest of Header																															

Source: wiki

2. What are TCP and UDP? What is the difference between them? Show the protocol header for each.
   2 different protocols define how to establish & maintain network communication.
   Tcp is connection-oriented protocol while udp is connectionless.
   Tcp is slower but does error recover. It is heavy weight. It has handshake protocols.
   Udp is faster but discards erroneous packets. It is light weight. It doesn't have handshake protocols.

Offsets	Octet	0								1								2								3							
Octet	Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	Source port																Destination port															
4	32	Sequence number																															
8	64	Acknowledgment number (if ACK set)																															
12	96	Data offset				Reserved 0 0 0				NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Window Size														
16	128	Checksum																								Urgent pointer (if URG set)							
20 ...	160 ...	Options (if *data offset* > 5. Padded at the end with "0" bytes if necessary.) ...																															

Tcp, source: wiki



UDP datagram header

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Source port																Destination port															
4	32	Length																Checksum															

udp, source: wiki

3. What is ARP and what is it used for?
   Address resolution protocol. It is a communication protocol to discover link layer address (e.g. MAC address) based on ip address.

4. Explain in detail what a TCP Three Way Handshake is. Show an illustration for the setup AND teardown process of a handshake.

   To start with, TCP is bi-direction.
   Three way handshake is the process of client & server try to establish communication.

   In order to establish stable connection, we need to check client & server the send & receive function is ok. That should be 4 times.
   C for client & S for server
   C->SYN->S (client send ok)
   C<-ACK<-S(server receive ok)
   C<-SYN<-S(server send ok)
   C->ACK->S(client receive ok)

   In order to optimize, we can see that 2 packets in the middle can be combined.
   C->SYN->S
   C<-ACK/SYN<-S
   C->ACK->S
   That's why it's called 3 ways.

   When terminate, both client & server should inform others I'm finished and make sure the other have received the signal. That's 4.
   C->FIN->S
   C<-ACK<-S
   C<-FIN<-S
   C->ACK->S
   Can we optimize? No. Because TCP is bi-direction and they may not finished their part. Therefore it's 4.

6 Question 6: Networking Protocols 8 / 8

✓ - **0 pts** **Correct**

- **0.5 pts** Incorrect networking tool (program) that uses ICMP
- **1 pts** No illustrations for TCP setup/teardown
- **0.5 pts** Never mentions that TCP/UDP are transport layer protocols
- **0 pts** TCP = TRANSMISSION control protocol
- **1 pts** No TCP/UDP headers
- **0.5 pts** Need more detail on ARP
- **1 pts** No teardown for TCP handshake
- **0.5 pts** No ICMP protocol header
- **0.25 pts** Does not address important differences between TCP and UDP
- **0.5 pts** Insufficient detail on ICMP
- **0.25 pts** Doesn't talk about teardown in TCP handshake
- **0.75 pts** Lacking most relevant details on TCP/UDP

Question 7: Ports (8 points)

1. Explain what a TCP/UDP port is and give an example.
   Port: the communication endpoint.
   Tcp/udp port: the communication endpoint based on the transmission protocols
   e.g. port 80 is for http

2. How many bits are in a port number?
   16 bits

3. How many ports numbers are there (what is the range)?
   0-65535

4. What organization is in charge of registering services with port numbers?
   IANA

5. What service commonly runs on the following TCP ports:
   a. 21 FTP control
   b. 22 ssh, scp, sftp, port forwarding
   c. 23 unencrypted text communication
   d. 25 smtp
   e. 53 dns
   f. 80 http
   g. 135 dce, dcom, epmap
   h. 139 NetBIOS session service
   i. 443 https
   j. 445 smb
   k. 993 imaps
   l. 1433 mssql server
   m. 3306 mysql database system
   n. 3389 windows based terminal

7 Question 7: Ports 8 / 8

✓ **- 0 pts Correct**

- 1 pts Port example?

- 0.25 pts Off by one error (port numbers)

- 1 pts Wrong organization (IANA)

- 1 pts Missing crucial part of port definition (application level multiplexing)

- 0.5 pts No port number range specified

Question 8: DNS (8 points)

1. Explain what DNS is.
   Domain name system. A mapping between ip address and human readable name. For example, we only type google.com rather than the ip address we want to access when we try to get some info from the server. DNS translate google.com to actual server ip.

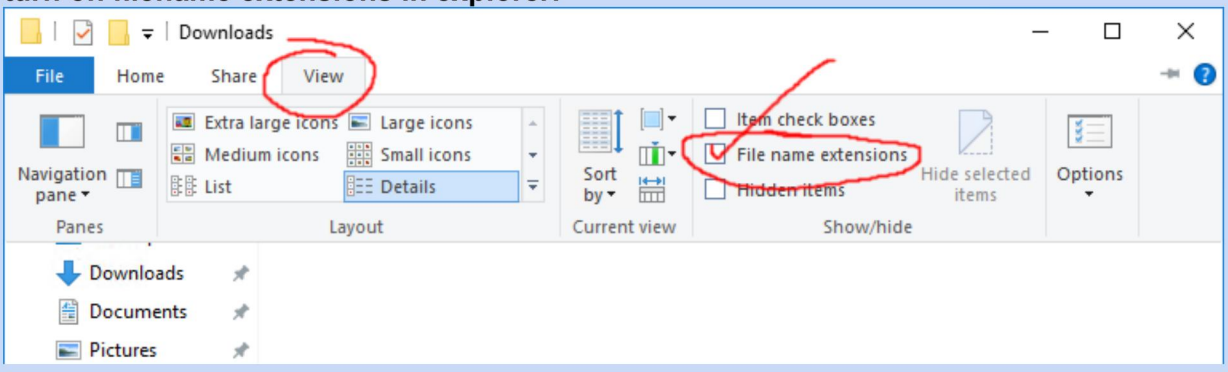
2. Name two programs you can use to get information from a DNS server.
   nslookup, dig

3. What is the default TCP/UDP port used by DNS?
   53

4. What is the domain for Duke and the subdomain for the ECE department?
   duke.edu
   ece.duke.edu

Quick side thing: Fix a dumb Windows security issue
We're about to use our Windows VM for the first time. By default, Windows does something mind-bogglingly stupid and bad: hiding filename extensions. If you're doing anything more with the computer than emailing grandma, this is infuriating, and can easily lead to security issues like the classic *masquerading EXE*: a malware "CatPicture.jpg.exe" will just show as "CatPicture.jpg", making the user think it's safe to run.

On your Windows VM (and on all Windows machines you touch until you die),
turn on filename extensions in explorer:



8 Question 8: DNS 8 / 8

✓ **- 0 pts** **Correct**

- 1 pts Looking for programs that query DNS, not pieces of the network

Question 9: Network Traffic Analysis with Wireshark (4 points)

Network analysis is the process of capturing network traffic and inspecting it closely to determine what is happening on the network. A network analyzer decodes, or dissects, the data packets of common protocols and displays the network traffic in human-readable format. Throughout this course we will be analyzing and inspecting a significant amount of network traffic. It is important that you become familiar with the tools that will allow you to capture and analyze network traffic. For this problem, we will be using a security tool called **Wireshark**.

Log into your Windows VM server. Download and install Wireshark. Use Wireshark to capture some network traffic on the public interface and display some contents of the traffic you captured.

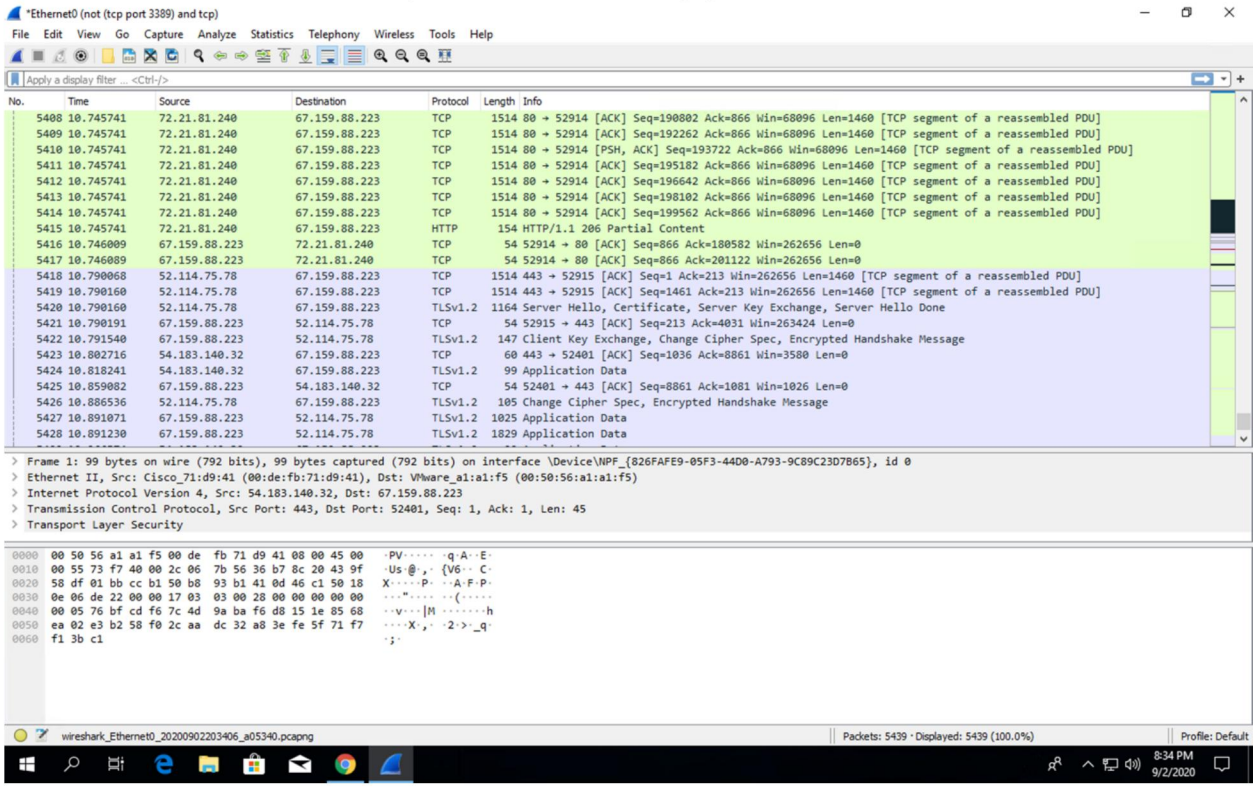
Notes:

- For capturing: Click Capture, Options, and click select interface with the public IP.
- Use a capture filter "not (tcp port 3389) and tcp"[1] on the selected network interface. This will filter out RDP traffic, which is how you're viewing the Windows GUI.
- Note: By default, you'll only be sniffing this machine's traffic. To do otherwise is to enter *promiscuous mode* which you should not do (it is both not ethical in this shared environment, and not likely to succeed given the network configuration).

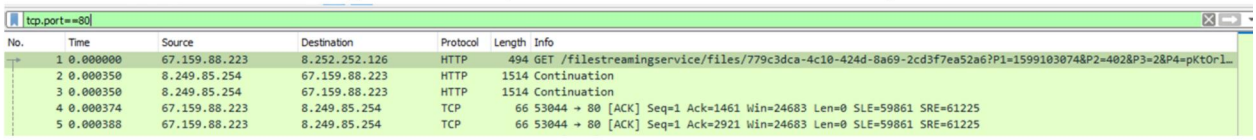
Your answer should include three pasted screenshots:

[1] Updated 2019-08-30: Filters updated for Wireshark 3.x

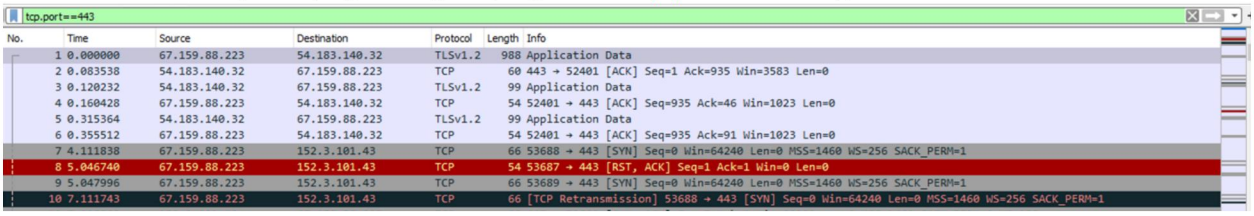
1. A screenshot of network traffic you didn't intentionally generate.



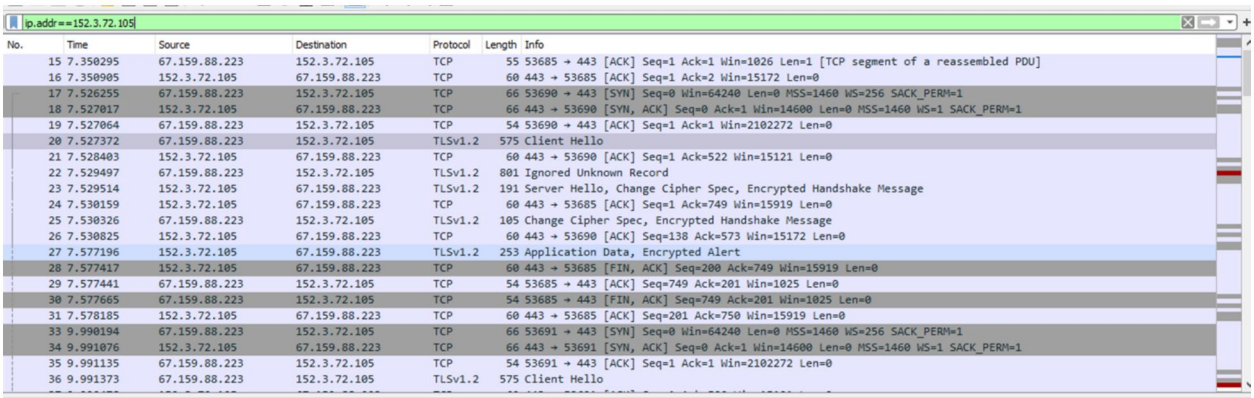
2. While the packet trace is running, open a browser and visit the course page at this URL: http://people.duke.edu/~tkb13/courses/ece560/
   Then, in Wireshark, stop the trace and find the HTTP request for the course site in the packet trace. You may use the display filter "tcp.port == 80" to make finding it easier. Take a screenshot showing the HTTP protocol details in the bottom pane.



3. Again, while a trace is running, open a browser and visit the course page at this URL: https://people.duke.edu/~tkb13/courses/ece560/
   Note that this URL is HTTPS instead of plain HTTP. Again stop the trace, find the HTTPS request (e.g. using display filter "tcp.port == 443"), and take a screenshot.



After filtering the dst server



Question: How much are you able to determine about the transaction in Wireshark in HTTP vs HTTPS?

1.we can see that they are both based on the tcp, only different is whether the payload is encrypted or not
2.in http, we can see the protocol for payload is http—meaning that content not encrypted
3.in https, we can see the protocol for payload is tlsv1.2—the current industry encryption standard, and from the content we can see the encryption process

9 Question 9: Network Traffic Analysis with Wireshark 3 / 4

- **0 pts** Correct

✓ **- 1 pts** **Incorrect HTTP request (Should see obvious info from course website, and the IP for the course site is 152.3.72.105)**

- **1 pts** The second snapshot should show HTTP protocol details, not TCP

- **1 pts** Incorrect HTTPS request (Should be same dest. IP as HTTP.) (For the course site, 152.3.72.105)

- **0.75 pts** Should be able to see source/dest IP for HTTPS, otherwise no way to verify

- **1 pts** The third snapshot should show HTTPS requests, not TCP. HTTPS appears to be TLSv1.2 in wireshark, since it can be referred to as "HTTP over TLS".

- **1 pts** Incorrect observation. The major difference is that the messages and type of requests are encrypted in HTTPS, which is not in HTTP.

- **0.75 pts** Incorrect observation. The major difference is that the messages and type of requests are encrypted in HTTPS comparing with HTTP.

Question 10: Network Traffic Analysis with TCPDump (3 points)

TCPDump is a common computer network debugging tool that runs under the command line. It allows the user to intercept and display TCP/IP and other packets being transmitted or received over a network to which the computer is attached.

Log into your Linux VM and use TCPDump to capture some network traffic and display the contents the traffic you captured.

Here is a command that will capture 10 packets using tcpdump:

```
$ sudo tcpdump -i eth0 -c 10
```

(Note: tcpdump was installed by default on my Linux VM. If it isn't for you, you can install it with "`sudo apt install tcpdump`")

```
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
21:12:51.421631 ARP, Request who-has 152.3.52.1 tell vcm-16667.vm.duke.edu, length 46
21:12:51.427597 IP vcm-16036.vm.duke.edu.49244 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 37670+ [1au] PTR? 1.52.3.152.in-addr.arpa. (52)
21:12:51.431414 IP vcm-16036.vm.duke.edu.ssh > 10.172.17.217.50726: Flags [P.], seq 3534533997:3534534233, ack 77105532, win 1959, options [nop,n
op,TS val 3918115162 ecr 780764011], length 236
21:12:51.432946 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-16036.vm.duke.edu.49244: 37670 NXDomain* 0/1/1 (131)
21:12:51.433022 IP vcm-16036.vm.duke.edu.49244 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 37670+ PTR? 1.52.3.152.in-addr.arpa. (41)
21:12:51.434568 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-16036.vm.duke.edu.49244: 37670 NXDomain* 0/1/0 (120)
21:12:51.434982 IP vcm-16036.vm.duke.edu.47939 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 14683+ [1au] PTR? 68.52.3.152.in-addr.arpa. (53)
21:12:51.439201 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-16036.vm.duke.edu.47939: 14683* 1/0/1 PTR vcm-16667.vm.duke.edu. (88)
21:12:51.439719 IP vcm-16036.vm.duke.edu.33896 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 13738+ [1au] PTR? 100.72.3.152.in-addr.arpa. (54)
21:12:51.441710 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-16036.vm.duke.edu.33896: 13738* 1/0/1 PTR rsv-bc-fitzcachedns.oit.duke.edu. (100
)
10 packets captured
17 packets received by filter
0 packets dropped by kernel
```

We will be using Wireshark and TCPDump among other network traffic analyzers very heavily throughout the semester. I recommend spending some time with these tools and learning some of the features they have to offer. You don't need to understand all the output of these packets right now, but as we spend more time with these tools you will learn to dissect the output and be able to find the information you are looking for.

10 Question 10: Network Traffic Analysis with TCPDump 3 / 3

✓ - **0 pts** **Correct**

Question 11: Network Mapping (7 points)

Nmap is a free and open source utility for network exploration or security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other features. It was designed to rapidly scan large networks, but works fine against single hosts. Nmap runs on all major computer operating systems, and both console and graphical versions are available.

Log into your Linux VM and install nmap from the package manager:

```
$ sudo apt install nmap
```

Use Nmap to port scan your Windows VM. Here is the command you should use[2]:

```
$ sudo nmap -p- -v -sT -Pn <TARGET_MACHINE>
```

Include in your answer the following:

1. Explain each parameter of this command.
   -p-: to scan ports from 1 through 65535
   -v: enables verbose mode
   -sT: TCP connect scan
   -Pn: Treat all hosts as online -- skip host discovery

   Source: man nmap

2. Paste the results of the scan.

[2] In command line explanations, items in <ANGLE BRACKETS> are required inputs and items in [SQUARE BRACKETS] are optional inputs. Either way, *don't include the brackets themselves!*

```
gc171@vcm-16036:~$ sudo nmap -p- -v -sT -Pn vcm-16210.vm.duke.edu
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-02 21:27 EDT
Initiating Parallel DNS resolution of 1 host. at 21:27
Completed Parallel DNS resolution of 1 host. at 21:27, 0.00s elapsed
Initiating Connect Scan at 21:27
Scanning vcm-16210.vm.duke.edu (67.159.88.223) [65535 ports]
Discovered open port 3389/tcp on 67.159.88.223
Discovered open port 135/tcp on 67.159.88.223
Connect Scan Timing: About 20.45% done; ETC: 21:29 (0:02:01 remaining)
Discovered open port 5986/tcp on 67.159.88.223
Connect Scan Timing: About 48.82% done; ETC: 21:29 (0:01:04 remaining)
Discovered open port 5040/tcp on 67.159.88.223
Discovered open port 2701/tcp on 67.159.88.223
Discovered open port 7680/tcp on 67.159.88.223
Discovered open port 5985/tcp on 67.159.88.223
Completed Connect Scan at 21:28, 104.50s elapsed (65535 total ports)
Nmap scan report for vcm-16210.vm.duke.edu (67.159.88.223)
Host is up (0.00045s latency).
Not shown: 65528 filtered ports
PORT     STATE SERVICE
135/tcp  open  msrpc
2701/tcp open  sms-rcinfo
3389/tcp open  ms-wbt-server
5040/tcp open  unknown
5985/tcp open  wsman
5986/tcp open  wsmans
7680/tcp open  pando-pub

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 104.54 seconds
```

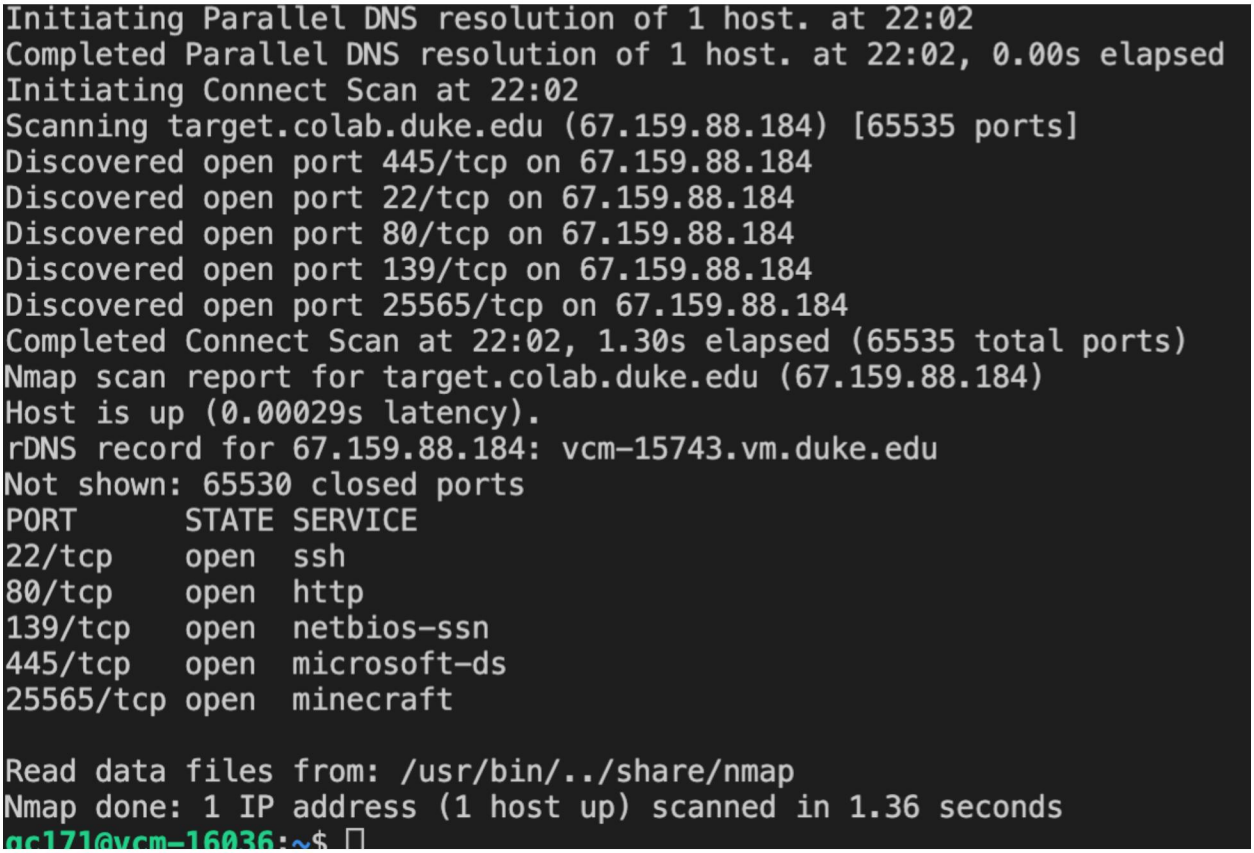
3. Note each port that is open and look up what service each corresponds to (not just the name of the service, but what it *accomplishes*).
   135/tcp msrpc the client/server model in Windows NT framework
   2701/tcp sms-rcinfo Microsoft system management server remote tools
   3389/tcp ms-wbt-server Windows based terminal for Windows remote desktop
   5040/tcp unknown
   5985/tcp wsman SOAP based protocol for the management of the servers
   5986/tcp wsmans SOAP based protocol for the management of the servers
   7680/tcp pando-pub pando media public distribution, to transfer large files

TIP: Read man pages (available via the command line and the web) for the various command line security tools to learn details about the different functions and parameters. Another useful tool for understanding command parameters is the website explainshell.

Next, let's scan an example Linux VM of mine, **target.colab.duke.edu**.

What network ports are open on this server?

Show output of the nmap scan and explain what services are running on the machine.



22/tcp ssh: for secure shell connection
80/tcp http: transmitting hypermedia document, such as HTML
139/tcp netbios-ssn: network basic i/o system, provides service to session layer, allowing computers at LAN to communicate
445/tcp microsoft-ds: microsoft directory service, windows use it for file sharing
25565/tcp minecraft: game

11 Question 11: Network Mapping 7 / 7

✓ **- 0 pts Correct**

- 0.5 pts Minor Parameter Explanation Error

- 1 pts Major Parameter Explanation Error

- 1 pts No Windows result shown

- 1 pts Minor Error on Explanation to service

- 0.5 pts Scan the wrong site

- 1 pts No Windows Service Explanation

- 0.5 pts Missing Screenshots

- 1 pts No target service explanation

- 3 pts Did not scan Windows VM

- 0 pts Do one scan and explanation correctly, and you get full credit.

https://piazza.com/class/kd6irc9vkbl23b?cid=61

Question 12: Ncat, Telnet, Netstat, and Sockets (12 points)

Part 1: Intro to some basic tools

One common thing to do is to use sockets "directly" (i.e., without much software in the way) to accomplish various networking goals. A common utility for this purpose is **netcat**. Netcat comes in two flavors: the classic **nc** (commonly pre-installed in many Linux distros) and a more modern rewrite called **ncat** that comes with nmap.

Both are in common use for completing many tasks involving TCP or UDP. They can open TCP connections, send UDP packets, listen on arbitrary TCP and UDP ports, do port scanning, and deal with both IPv4 and IPv6. The most common netcat command simply connects to a host on a given port and sends/receives data on stdin/stdout.

A related concept is the **telnet** tool and protocol. Telnet was the original way of connecting to a remote machine's shell like the way we use **ssh** today. Telnet is very simple: it basically just connects the stdin and stdout/stderr of the remote shell to a TCP socket. So when you type "ls", you're just sending an "l" and an "s" as bytes over a TCP connection, and the server is sending the ls output back to you over that same socket. This means that passwords and other material are sent unencrypted, which is why use of telnet is discouraged today. That said, telnet is shockingly alive and well in a variety of corporate and IoT environments because of how simple and inexpensive it is to implement. Further, the underlying notion of hooking a shell right up to a socket is sometimes used by attackers as a simple way to create backdoor access to a machine. The telnet tool itself can also be useful as it functions as a very simple "open a socket and let me type into it" tool, like a simplified netcat on machines where netcat is not installed.

In addition to making connections with the above tools, it is possible to query the OS to find out what connections are currently established system-wide. On both Linux and Windows, the command to do this is **netstat** (though the options differ between the two).

There are hundreds of uses for these utilities. In this assignment, we just want you to learn a couple of them.

On your Windows VM, download and extract the ZIP archive of nmap tools for Windows from here (*not the installer* -- we don't need a full installation, and an attacker wouldn't do one, as that creates more visible evidence of intrusion). Open a command prompt and navigate to where you extracted the tools. If using PowerShell as your prompt instead of the classic shell, you may need to prefix commands with `.\` (similar to `./` on Linux).

By running the ncat command from a command shell on a Windows Server box, anyone that telnets to port 4455 on that box would encounter a command shell without even having to login. Basically, this command starts a service on the current box that listens on port 4455 for incoming connections. This a common backdoor that attackers put on servers.

```
ncat -l 4455 -e cmd.exe
```

Open a command prompt and run the command. When you run the command it will appear to just hang. It is actually not hanging but listening on port 4455 for incoming connections. (Note: your Windows VM has a live internet-facing IP address, so do NOT leave this open for long -- move on to the next part so we connect to it. If you leave this listening, an automated attacker from the internet *will* connect to it and potentially take over the VM!)

On the Windows VM, open a second command prompt and run netstat to see the socket listening on port 4455 and **post a screenshot**:

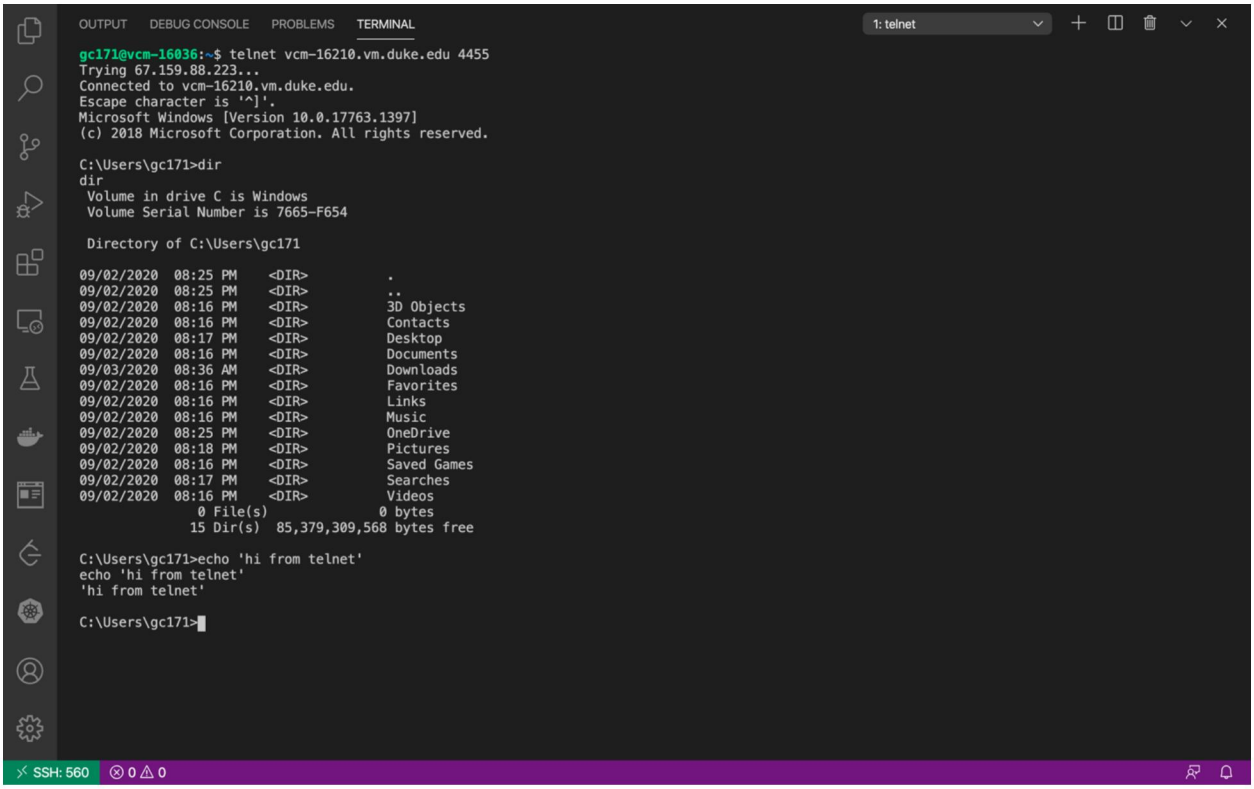
```
netstat -anop tcp
```

```
TCP    0.0.0.0:4455           0.0.0.0:0              LISTENING       5960
```

Now from your Linux VM, telnet into the Windows box to establish a connection. The following command will connect you to your Windows server via a telnet connection to port 4455.

```
telnet <WINDOWS_MACHINE_IP> 4455
```

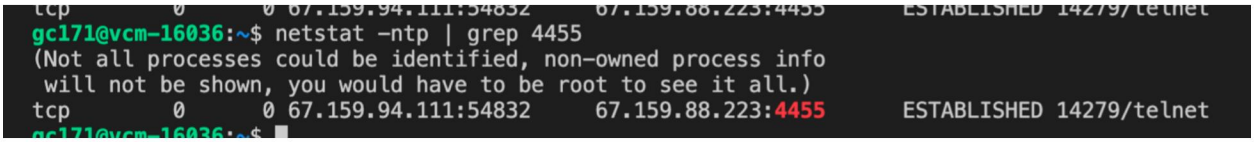
Some shell features won't work (e.g. up-arrow, cursor controls, etc.), but you should be able to run commands and see output. **Run some commands and post a screenshot** (be sure to show the initial telnet command in your screenshot so we can tell it worked).



Once you have received a command shell on the Linux VM, in a new separate command prompt, run the command:

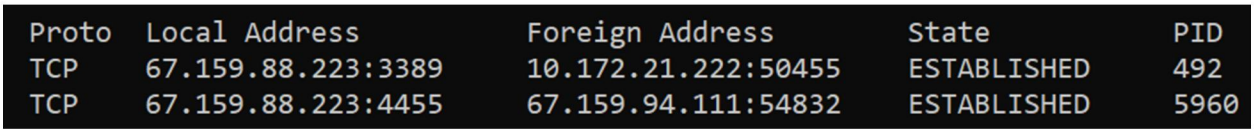
```
netstat -ntp
```

You should see your outgoing connection on Linux box to see your connection running on port 4455. **Post a screenshot.**



On the Windows machine via RDP, open a new command shell and run netstat to see the connection from that end and **post a screenshot**:

```
netstat -nop tcp
```



Close the command shell by typing *exit* to end the ncat service running.

Part 2: Catching a reverse shell

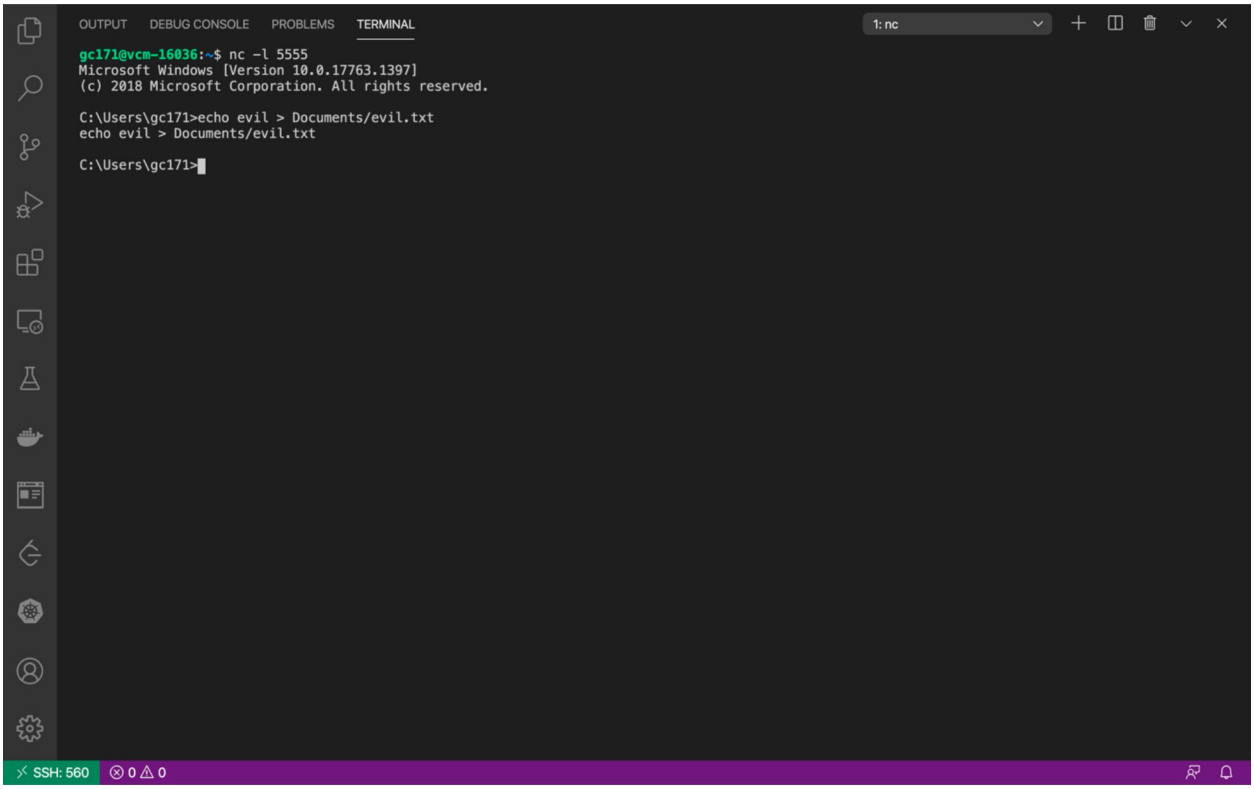
Often, an attacker will gain the ability to issue a command on a victim machine and will use that command to establish a foothold.

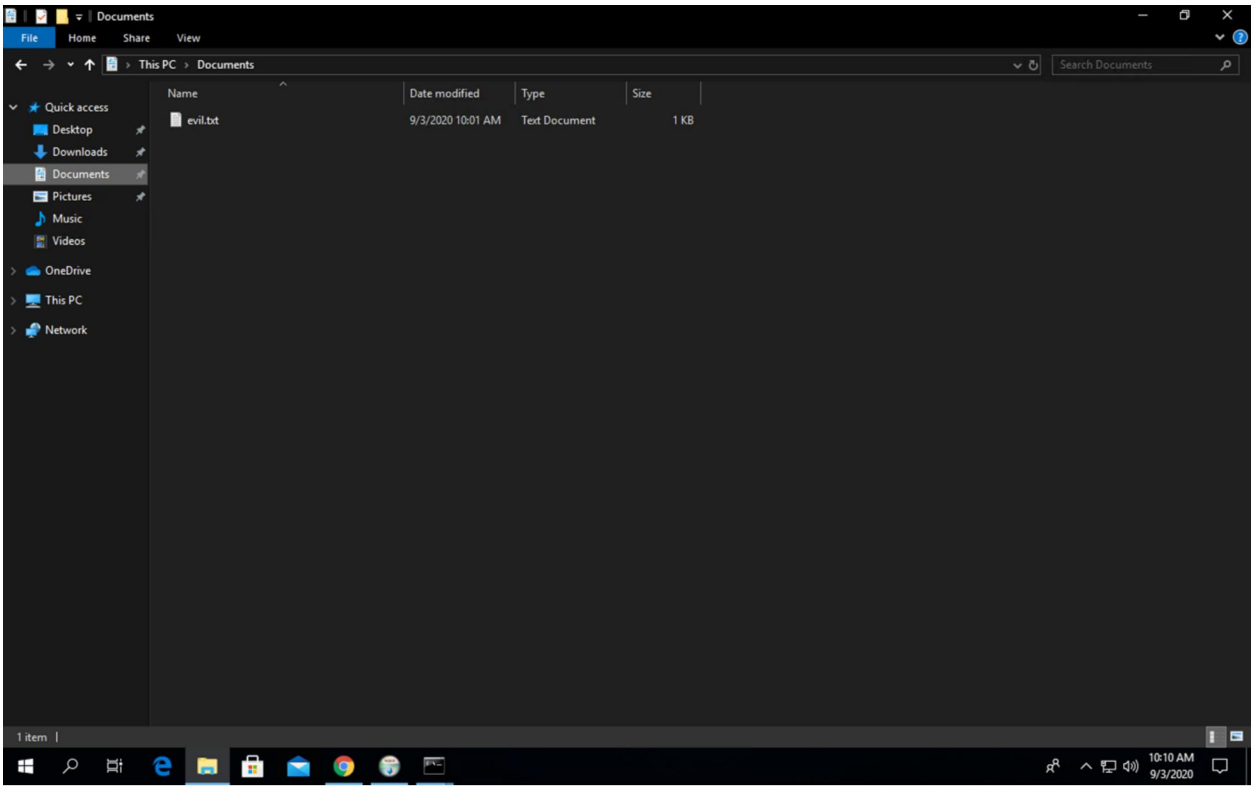
Well assume your Linux VM is the attacker machine. Use netcat (`nc`) to listen on a TCP port of your choice.

Your Windows VM will be the victim. Use netcat (`ncat`) to connect to your Linux VM on the specified port while executing a `cmd.exe` shell.

If successful, you should see a Windows command prompt appear on your Linux VM. This is called *catching a reverse shell*, and is a very common technique for attackers.

As a demonstration, using this command prompt, put a file called “evil.txt” into the victim’s Documents directory. Paste a **screenshot of your Linux console doing this** as well as a **screenshot of the Windows VM’s documents folder showing the evil document having been created**.




Documents
File
Home
Share
View
This PC
Documents
Search Documents
Quick access
Desktop
Downloads
Documents
Pictures
Music
Videos
OneDrive
This PC
Network
Name
Date modified
Type
Size
evil.txt
9/3/2020 10:01 AM
Text Document
1 KB
1 item
10:10 AM
9/3/2020

12 Question 12: Ncat, Telnet, Netstat, and Sockets 12 / 12

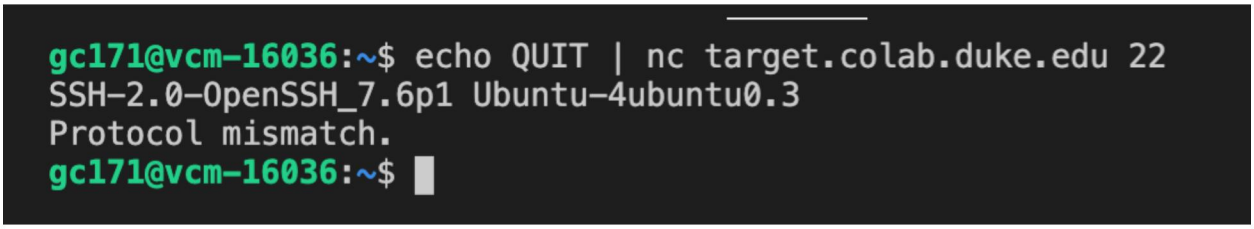
✓ - 0 pts Correct

Question 13: Banner Grabbing: Services Spilling Their Guts (8 points)

After using Nmap or another port scanner to identify what ports are open on a system, you may like to be able to get more information about those ports. You can usually accomplish this by connecting to a port; the service will immediately spill its version number, software build version, and perhaps even the underlying operating system.

For example, from your Linux VM, run this command and **post a screenshot of the output**.

```
echo QUIT | nc target.colab.duke.edu 22
```



To become better acquainted with sockets, you will write a small socket-based program called **getbanner** to do the above operation. You may write it in the language of your choice, but it must run (and compile, if using a compiled language) on a standard Linux environment such as the Ubuntu 18.04 of your Linux VM. The only further restriction is that it may not use telnet, ncat, or nc in its operation (otherwise, a bash script literally containing the snippet above would suffice, and that wouldn't be very interesting).

The algorithm for the program will be similar to the shell command shown above:

1. Get the hostname and port from the command line arguments.
2. (If none are supplied, print an appropriate usage message.)
3. Connect to the given host on the given TCP port.
4. Send the remote host the string "QUIT\n".
   (This isn't a standard -- some protocols recognize this as a legitimate quit command, and for those that don't, most will print their version information regardless of what the clients send.)
5. Read everything the server[3] sends, printing it to the console as it's received.
6. When the server disconnects, quit.

Note: this is just 10-30 lines of code, depending on language (even in Java).

Submit a zip file called `<netid>_getbanner.zip` with your code and a Makefile (if needed) to the Sakai locker for this assignment.

NOTE: You are submitting the **zipped code** to **Sakai** and the **PDF answers** to **Gradescope**.

[3] Updated 2019-09-13: This used to say "client", which was a mistake.

13 Question 13: Banner Grabbing: Services Spilling Their Guts 8 / 8

✓ **- 0 pts** **Correct**

- 1 pts Missing screenshot for the command

- 6 pts No zip file submitted to Sakai

Question 14: Networking Tools (9 points)

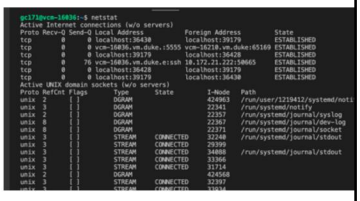
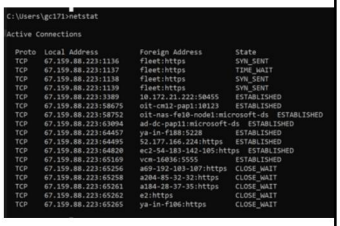
Linux and Windows have lots of networking tools that are built into the operating system. These tools are very valuable to know and understand because they become very useful for troubleshooting, system forensics, network assessment, etc. These are not classified as security tools, but most security professionals use them on a daily basis.

For both a Linux-based system and Windows-based system, learn to use the following commands: netstat, ifconfig/ipconfig, nslookup, traceroute/tracert, ping, pathping, host, dig, top, ps/tasklist.

For help on Linux commands, type “man toolname” (example, “man ping”)
For help on Windows commands, type “toolname /?” (example, “ping /?”)

The reason you are learning these tools for both operating systems is because some of the flags/switches for these tools differ between them and even versions of the OS.

For each of the tools below, fill in the table with the system information and a brief description of the tool.
For any utility that requires a hostname use *duke.edu*
Use your Windows and Linux VMs for this exercise for consistent output.

TOOL	Brief Description	Brief Linux output	Brief Windows output
netstat	Print network info(protocol, connections		
ip (Linux) ipconfig (Windows)	Show network interface, routing	$ ip address 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00 inet 127.0.0.1/8 scope host lo valid_lft forever preferred_lft forever inet6 ::1/128 scope host valid_lft forever preferred_lft forever 2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000	>ipconfig Windows IP Configuration Ethernet adapter Ethernet0: Connection-specific DNS Suffix . : Link-local IPv6 Address . . . . . : fe80::59d0:9381:cb7:5b0b%5 IPv4 Address. . . . . : 67.159.88.223 Subnet Mask . . . . . : 255.255.254.0 Default Gateway . . . . . : 67.159.88.1 Ethernet adapter Npcap Loopback Adapter:

		link/ether 00:50:56:a1:87:25 brd ff:ff:ff:ff:ff:ff inet 67.159.94.111/23 brd 67.159.95.255 scope global eth0 valid_lft forever preferred_lft forever inet6 fe80::250:56ff:fea1:8725/64 scope link valid_lft forever preferred_lft forever	Connection-specific DNS Suffix . : Link-local IPv6 Address . . . . . : fe80::c5f3:d804:1e51:9ba2%11 Autoconfiguration IPv4 Address. . : 169.254.155.162 Subnet Mask . . . . . : 255.255.0.0 Default Gateway . . . . . :
nslookup	Query internet domain name server	$ nslookup duke.edu Server: 127.0.0.53 Address: 127.0.0.53#53 Non-authoritative answer: Name: duke.edu Address: 152.3.72.104	>nslookup duke.edu Server: rsv-bc-fitzcachedns.oit.duke.edu Address: 152.3.72.100 Name: duke.edu Address: 152.3.72.104
traceroute (Linux) tracert (Windows)	Print route packets trace to host	$ traceroute duke.edu traceroute to duke.edu (152.3.72.104), 30 hops max, 60 byte packets 1 * 152.3.53.254 (152.3.53.254) 0.665 ms 0.843 ms 2 * * * 3 10.236.254.238 (10.236.254.238) 0.738 ms 10.236.254.226 (10.236.254.226) 0.969 ms 1.210 ms 4 tel1-sp-resnet-vrf-v4309.netcom.duke.edu (10.236.242.114) 1.318 ms 1.491 ms 1.921 ms 5 10.236.244.121 (10.236.244.121) 1.495 ms 1.799 ms 1.625 ms 6 10.236.254.227 (10.236.254.227) 1.442 ms 1.596 ms 1.706 ms 7 fitzeast-white-dc-nx-po50.netcom.duke.edu (10.237.254.1) 9.195 ms fitzeast-white-dc-nx-po51.netcom.duke.edu (10.237.254.3) 9.694 ms 9.571 ms 8 duke-web-fitz.oit.duke.edu (152.3.72.104) 1.250 ms 0.915 ms 1.158 ms 9 fitz-ltm-01-ex.oit.duke.edu (152.3.72.251) 4837.562 ms !H 4835.514 ms !H *	>tracert duke.edu Tracing route to duke.edu [152.3.72.104] over a maximum of 30 hops: 1 <1 ms <1 ms <1 ms 152.3.53.254 2 * * * Request timed out. 3 <1 ms <1 ms <1 ms 10.236.254.226 4 1 ms 1 ms 1 ms tel1-sp-resnet-vrf-v4309.netcom.duke.edu [10.236.242.114] 5 1 ms 1 ms 1 ms 10.236.244.121 6 1 ms 1 ms 1 ms 10.236.254.227 7 1 ms 1 ms 1 ms fitzeast-white-dc-nx-po51.netcom.duke.edu [10.237.254.3] 8 1 ms 1 ms 1 ms duke-web-fitz.oit.duke.edu [152.3.72.104] Trace complete.
ping	send ICMP ECHO_REQUEST to network hosts	$ ping duke.edu PING duke.edu (152.3.72.104) 56(84) bytes of data. 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=1 ttl=248 time=0.897 ms	>ping duke.edu Pinging duke.edu [152.3.72.104] with 32 bytes of data: Reply from 152.3.72.104: bytes=32 time=1ms TTL=248 Reply from 152.3.72.104: bytes=32 time=1ms TTL=248

		64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=2 ttl=248 time=1.08 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=3 ttl=248 time=1.10 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=4 ttl=248 time=1.14 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=5 ttl=248 time=2.70 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=6 ttl=248 time=1.48 ms	Reply from 152.3.72.104: bytes=32 time=1ms TTL=248 Reply from 152.3.72.104: bytes=32 time=1ms TTL=248 Ping statistics for 152.3.72.104: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 1ms, Maximum = 1ms, Average = 1ms
pathping	Combination of tracert & ping	N/A	>pathping duke.edu Tracing route to duke.edu [152.3.72.104] over a maximum of 30 hops: 0 vcm-16210.win.duke.edu [67.159.88.223] 1 152.3.53.254 2 * * * Computing statistics for 25 seconds... Source to Here This Node/Link Hop RTT Lost/Sent = Pct Lost/Sent = Pct Address 0 vcm-16210.win.duke.edu [67.159.88.223] 0/ 100 = 0% | 1 12ms 0/ 100 = 0% 0/ 100 = 0% 152.3.53.254 Trace complete.
host	DNS lookup`	$ host duke.edu duke.edu has address 152.3.72.104 duke.edu mail is handled by 10 mx.oit.duke.edu.	N/A
dig	DNS lookup	$ dig duke.edu ; <<>> DiG 9.16.1-Ubuntu <<>> duke.edu ;; global options: +cmd ;; Got answer: ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11830 ;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1	N/A

		;; OPT PSEUDOSECTION: ; EDNS: version: 0, flags:; udp: 65494 ;; QUESTION SECTION: ;duke.edu. IN A ;; ANSWER SECTION: duke.edu. 246 IN A 152.3.72.104 ;; Query time: 4 msec ;; SERVER: 127.0.0.53#53(127.0.0.53) ;; WHEN: Thu Sep 03 11:13:25 EDT 2020 ;; MSG SIZE rcvd: 53	
ps (Linux) tasklist (Windows)	Snapshot of current processes	$ ps PID TTY TIME CMD 14281 pts/2 00:00:00 bash 15187 pts/2 00:00:00 ps	>tasklist Image Name PID Session Name Session# Mem Usage ========== ======== ================ =========== ============ System Idle Process 0 Services 0 8 K System 4 Services 0 156 K Registry 88 Services 0 33,912 K smss.exe 364 Services 0 1,012 K csrss.exe 472 Services 0 4,068 K wininit.exe 544 Services 0 5,156 K csrss.exe 556 Console 1 3,368 K winlogon.exe 640 Console 1 6,556 K services.exe 672 Services 0 9,060 K lsass.exe 692 Services 0 20,132 K svchost.exe 796 Services 0 3,264 K fontdrvhost.exe 804 Services 0 2,548 K fontdrvhost.exe 812 Console 1 2,384 K svchost.exe 880 Services 0 24,504 K svchost.exe 932 Services 0 14,940 K

			svchost.exe 968 Services 0 9,560 K LogonUI.exe 428 Console 1 45,344 K svchost.exe 492 Services 0 68,504 K dwm.exe 464 Console 1 27,892 K
Example ping	Ping – send ICMP ECHO_REQUEST packets to network hosts	$ ping duke.edu PING duke.edu (152.3.72.197) 56(84) bytes of data. 64 bytes from 152.3.72.197: icmp_seq=1 ttl=240 time=21.7 ms 64 bytes from 152.3.72.197: icmp_seq=2 ttl=240 time=27.3 ms ^C --- duke.edu ping statistics --- 2 packets transmitted, 2 received, 0% packet loss, time 1001ms rtt min/avg/max/mdev = 21.728/24.554/27.380/2.826 ms	> ping duke.edu Pinging duke.edu [152.3.72.197] with 32 bytes of data: Reply from 152.3.72.197: bytes=32 time=27ms TTL=240 Reply from 152.3.72.197: bytes=32 time=22ms TTL=240 Reply from 152.3.72.197: bytes=32 time=25ms TTL=240 Reply from 152.3.72.197: bytes=32 time=22ms TTL=240 Ping statistics for 152.3.72.197: Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 22ms, Maximum = 27ms, Average = 24ms

~ END ~

14 Question 14: Networking Tools 9 / 9

✓ - 0 pts Correct

15 Late Penalty 0 / 0

✓ - 0 pts No penalty