Justin Strelka

Nmap\_report

6/7/20

**Mapping A Home Network**

**Abstract:**

This paper begins with a brief description of what it means to map a home network. Following this brief description there is an overview of the NMAP tool and how it is useful for mapping a network and why you need to be careful when deploying a network scan with NMAP or any other tool. Next, the paper will analyze a scan of a sample home network and explain the information NMAP provides. This analysis includes discussion about each service NMAP returns for a device and the protocol used by a service to communicate with its destination point. Lastly, the paper will conclude with a brief overview of steps to take for securing services on your LAN.

**Motivation:**

This paper is intended for individuals that are inexperienced with networking and have little to no prior knowledge of the tool NMAP. Upon reading this paper you will receive a baseline understanding how a home network communicates and be able to define fundamental networking terms such as IP addresses, ports and various communication protocols. Most importantly you will learn how to read the output of NMAP. You will be able to identify how many devices are connected to a network, what ports are being used, and the outbound services alongside their protocol. Understanding the output of a network scanning tool such as NMAP will assist you in identifying threats to a network and assist you in building/maintaining networks in the future.

**Introduction:**

In modernized countries having a local area network (LAN) for your home or small business has become an essential utility, yet most of us fail to understand the vulnerabilities involved in owning a LAN. However, there are some tools that exist free of charge that can assist us in better understanding how our networks work and what kind of communication happening on our networks. For the purpose of this paper we will be using the opensource tool NMAP to help us analyze a sample LAN with multiple devices running multiple services on the LAN. The NMAP utility provides us with a wide variety of information such as: devices are currently connected to our LAN, the operating systems running on each device, device services and their port number, and the topology of the devices on the LAN.

**Method/Measurement:**

Before we begin using the NMAP utility to analyze our sample network there is some terminology we must first understand that will help us decipher the technical output the NMAP utility will provide. First and foremost, “host” refers to a device discovered on your LAN. A host can be any device that can connect to your network. For example, a host can obviously be a laptop that is connected via wi-fi or ethernet cable. A host can also be a wi-fi/ethernet enabled printer since they too communicate across your LAN using various services that communicate through various protocols on defined ports. A service is simply a piece of software that performs automated tasks and when those automated tasks require communication from outside of the localhost the communication happens on ports. A port defines a specific communication channel within a host. When a router receives a packet/datagram that is to be routed to a specific host, the router looks at both the destination IP address and the port number. This combination of IP and Port number is called a socket and the router then sends the datagram to the specified socket destination point. When the datagram is received by the host with the specified IP address the host reads the port number and the service running on that port may or may not speak the same communication protocol as the packet. Lastly, a protocol is a type of communication language that both sides of the transmission can understand. Protocols can be both secure and unsecure and we need to be mindful of when unsecure protocols are being used across our LAN.

Now that we have a baseline understanding of the information the NMAP utility will provide us with and what it refers to, let’s move forward with installing the tool and begin analyzing a sample network. As mentioned NMAP is a free opensource utility designed for network discovery and security auditing that can be downloaded through the following link: <https://nmap.org/download.html>. Choose the correct installer for your machine, for the purposes of this paper the latest stable windows version 7.80 was used. It is recommended to give permission for all components to be installed so you don’t remove any desired functionality that you may later wish to enable.

Upon completing a successful install, we want to begin by opening the GUI for NMAP called Zenmap. This GUI can be very useful for perfecting the scan you wish to perform on your network and allows you to easily save previous scans into .xml files. Being able to save the contents of previous scans can be effective for understanding the effect changes to hosts on your network have on vulnerability. You can also look for abnormalities by having a healthy checkpoint to compare against if you begin to have problems with your LAN. Zenmap also organizes the data provided by NMAP into sub-categories that help us understand the meaning of all the information provided by NMAP and makes the information easy to find for humans.

We are going to begin with conducting a “ping scan” on our LAN to see how many hosts are currently connected. To conduct any kind of scan with NMAP we must first know the IP address range in which we wish to search for available hosts. It is very important that we understand exactly what IP addresses we will be scanning because an NMAP scan provides information that can be used to harm a network. We should never scan any network that we do not have permission to do so, therefore we recommend conducting theses tests on your own home LAN. To obtain the range of IP addresses we wish to scan on our home LAN we must first find the IP address assigned to your working device and the netmask. We can obtain this information using a variety of method however, I will be using the windows subsystem Linux (WSL) bash shell for the purposes of this test. The commands used will be native Linux bash shell commands which are different from the standard windows PowerShell and command prompt commands.

We can open the WSL command-line interface (cli) by navigating to the search bar next to the windows application viewer icon. Type in “wsl” and you should see an option for “wsl run command”, open it. Once the cli has opened we are going to use the command ifconfig and hit enter. You should see an output that looks like the one shown in Figure 1. The blue boxes outline the current active network interfaces. You can see that on that this device currently has four active network interfaces eth1, eth2, lo, and wifi0. The eth# interfaces are generally wired connections through a network interface card (NIC). The wifi# interfaces represent wireless connections happening on your device. Lastly, the “lo” interface is a local loopback that the device uses to communicate with itself. A loopback interface may be used for diagnostics and to connect to servers running on the local device [1]. This means communications over a loopback interface never leave the machine and are carried out by the devices operating system (OS) locally.

A screenshot of a cell phone

Description automatically generated

**Figure 1:** WSL ifconfig output from sample network

We can see the red boxes from **Figure 1** outline the inet field. The inet field displays the IPv4 address of the network interface or connection. This address tells the router where this device is on the network. On a LAN each IP should be unique and specify a single location on the network to send packets. If this number is not unique that means something with your router or network configurations is wrong or its possible that a malicious user is spoofing their IP address to intercept datagrams. The green boxes in **Figure 1** highlight the netmask field. The netmask field is used during routing to separate the network portion of the IP address from the host portion [2]. The network mask is essentially an “and” mask that operates on the IP address. We can see that my “wifi” interface has a netmask that is 24 bits long which signifies that this netmask is of the “class C” order. Meaning that the upper 24 bits of the IP address are used for network and the lower order 8 bits will be used for host enumeration. This means on a class C netmask we can have up to 255 visible hosts. Since netmasks are separated on the byte order this can cause an issue when trying to fine tune your network to the appropriate number of hosts. To solve this problem, we can also implement a technique called CIDR which allows us to divide the network portion from the host portion of our IP address by the bit. However, on a LAN which we are analyzing there will be no need for routing as routing only happens when communication is destined to go outside of a device’s LAN. Within the device’s LAN communication is established by a devices MAC address paired with the IP address. The MAC address is outlined by the purple box in **Figure 1.** The MTU value provided by the output of ifconfig refers to the size of datagram that the interface is configured to handle [3]. If a datagram is larger than the specified value, then the IP layer of the networking model must divide the datagram into subsections and transmit multiple packets of the specified MTU value or less [3]. Lastly, we can see an interface broadcast address outlined in yellow. The broadcast address is used when a device must send a signal to all devices on its current LAN. Each network interface has a default broadcast address configured which should be the network portion of the IPv4 address followed by all ones in a 32-bit structure. This makes sense that the loopback interface would not have a broadcast address since the loopback connection refers to the communication path within a device.

Now that we understand the ifconfig output, we can identify the IP address range we wish to scan using the NMAP GUI called ZENMAP. For the purpose of this paper we will be using IPv4 to scan our sample LAN. All the devices on the sample LAN are connected via wifi which means we need to locate the wifi# interface in the ifconfig output. We are interested in two values, the inet address and the network mask. Here we can see in **Figure 1** the inet address for the wifi0 interface is “192.168.0.2” and the network mask is “255.255.255.0”. We can see the network mask is of class C for this interface meaning the high order 24-bits refer to the network and the lower order 8-bits refer to the host. We now clearly identify that the lowest octet is the only octet reserved for visible devices on this network. This means that the IPv4 range we want to scan will be from 192.168.0.0 – 192.168.0.255.

A screenshot of a cell phone

Description automatically generated

**Figure 2:** zenmap GUI ping scan output of sample LAN

The ZENMAP GUI is can be used to visualize the NMAP utility so it is more user friendly than the CLI version of NMAP. In **Figure 2** we can see a target range of 192.168.0.0-255 which will scan every possible address a visible host could be on in our sample LAN. The ZENMAP GUI allows us to enter our IP range in the “Target:” field and choose what type of scan we wish to conduct in the “Profile:” field. After entering in our IP range and scan type we are presented with the CLI equivalence command that can be used on machines without graphics support or in a network report script [4]. **Figure 2** is a display of a “ping scan” which displays the number of visible hosts on the network, their MAC address, and topology of the network. Ping scans are useful for finding how many devices are on a network and since we can find the MAC address of visible devices, we can find out the device manufacturer [5]. However, if a device’s MAC address was changed from the manufacturer’s original address it would be ineffective to try and lookup the device manufacturer. As you can see a ping scan only sends ping requests to a target IP and does not provide us with port or communication protocol information.

The next scan we will look at is an NMAP quick scan which will find all visible hosts within the specified range. A quick scan will also provide us with information regarding the 100 most common TCP service ports such as the service running on a specified port. We can see in **Figure 3** that a quick scan on our sample LAN shows all visible hosts along with their IP and MAC address. Under the Ports/Hosts tab of a quick scan you will find information regarding the service and protocol being used to communicate on the 100 most common ports. I will describe in detail the services and protocols running on each port in the next scan type. The last thing to note about a quick scan is that it does not provide us with any host OS information, nor does it provide us with version information for any of the services running on the 100 most common port numbers.

**A screenshot of a cell phone

Description automatically generated**

**Figure 3:** ZENMAP quick scan ports/hosts tab screen

A screenshot of a social media post

Description automatically generatedThe final and most in depth scan we are going to look at is labeled “Intense scan.” This scan provides us with information regarding visible hosts within the specified range like a ping scan. An intense scan differs from a quick scan because it searches the 1000 most common ports used on each visible host within the specified range and provides the version of a service running on a port. Lastly, the Intense scan attempts to identify the OS of a host and provide us with the type of OS along with its version. **Figure 4** displays the detailed output of the NMAP utility as the intense scan progresses through the specified IPv4 range we provided (192.168.0.0-255).

**Figure 4:** Detailed output of intense scan on sample network

A screenshot of a cell phone

Description automatically generated

**Figure 5:** Host details view of 192.1680.1

**Results:**

The first of three hosts we will walk-through is at IPv4 address 192.168.0.1. We can see in **Figure 5** the host selected on the left-hand column is indeed 192.168.0.1 and the tab opened is “Host Details.” This device is a home router from CenturyLink. Under the host details tab, we can see various information regarding the selected host. Under the “Host Status” section we can see the state of the host which is currently up. We can see there are currently 3 open, 5 filtered, and 992 closed ports which represent the 1000 most commonly used ports scanned by “Intense Scan.” An open port means that a service on the host is listening for datagrams. A filtered port is a port means one of two things, either the port has a firewall that is blocking NMAP’s communication or the port is currently closed. Closed ports are not currently listening for any incoming traffic. However, closed ports do have the ability to open if configured to do so. If we look further down the tab, we can see there is an Operating System subsection which is where we can find information regarding the OS and its version currently running on the host. The host at 192.168.0.1 is currently running Linux 2.6.32-3.10. Knowing the operating systems of each device on our network will help us identify possible vulnerabilities to our network including those inherited with OS releases and/or verifying outdated versions. Knowing the OS of a host will also help us create effective firewalls and understand why particular services are running on the host.

Next we will look at the Ports/Hosts tab on ZENMAP for the Linux host at 192.168.0.1. We can see in **Figure 6** there are indeed 3 open ports and 5 filtered ports on the host. Port number 20 is a filtered port running ftp-data over TCP. TCP is a communication protocol that guarantees the sender will be notified if a datagram was received. This guarantee is done by a 3-way handshake between the sender and the receiver. The first datagram from the sender has metadata referring to a packets sequence number. The receiver then sends an acknowledgement packet back to the sender with metadata including the received sequence number and supplies the initial sender with an acknowledgement number. Upon receiving the acknowledgement packet the initial sender then sends yet another packet to the receiver with metadata including the incremented sequence number and the received acknowledge metadata. After this 3-way handshake has been completed both the sender and receiver have been synchronized. Now when the sender sends packets the receiver will know if a packet in the sequence has been dropped and upon a specified timeout can notify the sender that a packet in the sequence has not been received. Therefore, guaranteeing that the initial sender will be notified if there are dropped packets during transmission.

Now that we understand how TCP communication is established, we can begin discussing the FTP/FTP-data protocols. FTP is short for file transfer protocol and it is used to transfer and receive files between hosts. Transferring files between hosts requires users to authenticate on both sides of the transmission. On the sending side of the transmission users must have privileges to access the files for transfer. When the user enters the command to make a connection to the receiving end the user will be prompted for authentication credentials as well. Authentication on the receiving end of the transmission can be disabled, however disabling authentication should only be done on trusted local servers with trusted users. FTP uses two separate channels for establishing connections between client and server. The first channel is the command channel and displays as “ftp” in the NMAP utility. Port 21 is the default port that FTP listens to on the server side. The channel on port 21 is used for sending and receiving instructions between the client and server. The second channel used in an FTP connection is referred to as the data channel. The FTP-data channel runs on default port 20 on the server side and the main purpose of the FTP-data channel is to transmit file data between server and client. We can see in **Figure 6** that our host is indeed listening for FTP/FTP-data on ports 21/20 respectively. Lastly, both ports 21/20 are in a filtered state, meaning there is most likely a firewall set up to secure the device. If ports 21/20 were in an open state, the host would be vulnerable to attacks from outside the network from FTP over TCP.

The next protocol running on host 192.168.0.1 shown in **Figure 6** is secure shell (SSH) which is also a filtered connection running on the default port 22. Secure shell is a protocol that allows users to directly communicate with the command line shell of a remote server. Secure shell uses cryptography to secure transmissions between client and server [6]. When establishing an SSH connection users must authenticate to gain access to the shell on the server. Authentication can be done using credentials such as username and password or through use of public/private SSH keys. SSH keys are created as pairs containing both a public and a private key. The public key can be shared freely to remote hosts, whereas the private key must be kept safe and hidden to ensure the key has not been compromised. Authenticating using SSH keys the user must first copy their public key onto the remote host within their home directory at ~/.ssh/authorized\_keys. The host is then notified by the client to search for the specified public key within the authorized\_keys directory. The host will then create and encrypt a message using the public key and send it to the client and wait for a response from the client. If the client responds with the correct message decrypted along with a previously negotiated session id, the host knows the client has the authorized private key [6]. At this point the user will have gained access to communicate with the host through a secure shell.

Communication through the shell is also encrypted and requires the user to interact with a local shell that is a representation of the shell on the remote server. When the user enters a command, the transmission is encrypted and sent to the remote server. Once the message is received it is decrypted and processed through the shell. Output from the remote shell is then encrypted and sent back to the client for decryption and display to the user. The encryption keys for message communication are created using three unique inputs called symmetric keys. The three inputs used to create a symmetric key are the local private and public keys along with the remote server’s public key [6].

In **Figure 6** we can see a protocol called TELNET running on filtered port number 23. TELNET is a protocol that allows the user to connect to a remote server over TCP/IP. The TELNET protocol is like SSH in that it allows remote connection to the command line interface on a remote server, however TELNET lacks the encryption aspect of SSH [7]. Users must first authenticate using TELNET to secure a connection, though all commands are transmitted in plain text. This means all usernames, passwords, etc. are all visible during transmission and if these packets get intercepted the data in them is readily available to the attacker [7]. The only time TELNET should be used is when SSH is not available on the remote host.

Next we see Hyper Text Transfer Protocol (HTTP) over TCP/IP is listening on default port 80 in **Figure 6.** HTTP is the standard internet protocol used for surfing the web. HTTP is designed as a “pull protocol” meaning the client pulls information from a remote server. This pull style protocol works by the client sending a request for data from a remote server. The remote server then searches for the requested data and sends the data back to the client [8]. This type of design is like the master/slave metaphor where the client is the master and gives an order, then the remote server is the slave and fulfills the master’s requests. With this model all requests begin at the client level. HTTP is a stateless protocol meaning each request is independent of all previous requests. Neither side of the connection needs to track previous requests for future communication [8]. HTTP requests follow the “Method + Resource Path + Protocol version” format [8]. The “Method” is the action to be performed such as a **GET** or **POST** method. The various methods HTTP supports can be found here: <https://docs.oracle.com/cd/E19182-01/821-0540/ug_restbd-methods_c/index.html>. The resource path is simply the path to the resource on the server, like a file path on your local machine. The final portion of an HTTP request is the “Protocol version” which tells the server the http version the client will use to process the returned packet.

HTTP on port 80 serves as a redirect for incoming traffic to HTTPS which is ran at default port 443 [9]. Therefore, we see the unsecure protocol HTTP in an open state running on port 80. Most of the time when a user lands on a webpage using HTTP through port 80 the server will redirect the traffic to HTTPS which is a secure version of HTTP [9]. Again, like the difference between TELNET and SSH the data sent by HTTP is in plain text without any encryption. Meaning anybody who intercepts the packets can easily read the contents of each packet and extract sensitive data. HTTPS works the same way as HTTP but adds a level of security to the packets by first encrypting them at the transport layer [9]. With the rise of ecommerce websites, it is very important that our purchasing information is sent over HTTPS so attackers cannot easily read any personally identifiable information.

The TRAM protocol running on filtered port 4567 in **Figure 6** is used for bulk data transfer. The acronym TRAM stands for “Tree-Based Reliable Multicast Protocol.” This protocol is highly efficient for sending significant amounts of identical data from a single sender to multiple clients or even the entire LAN [10]. For a single client to create and send a separate copy of identical transmissions to multiple hosts can be a costly procedure. The TRAM protocol supports a tree-based hierarchy to allow for reliable and scalable communications. Trees are composed of host groups and each host group has a designated receiver. The receiver caches all messages sent by the sender and provides recovery assistance to any host that reports missing packets in a sequence like TCP [10]. By using the TRAM protocol, the entire network receives less traffic and requires less resource consumption by the sending host.

The last service running on host 192.168.0.1 is Universal Plug and Play (UPNP) on open port number 5341. UPNP allows multiple devices such as laptops, mobile devices, printers, wifi, etc. to discover each other and establish connections to share services and data. UPNP is designed to allow direct networking between devices on a LAN. UPNP assumes that devices understand standard IP addressing and uses various protocols such as TCP, HTTP, etc. that are built on standard IP addressing [11]. UPNP devices such as routers use SSDP to for discovery of other hosts on the LAN. UPNP is useful in residential networks because it allows for devices to communicate easily with no prior knowledge by the user making it ideal for device manufacturers to reduce initial set-up service calls. However, UPNP does have some inherent risks involved with its ease of use. To communicate over UPNP there is no authentication required which means any service running on a device can forward transmissions using a UPNP port [11]. This means that any malicious program that may have found its way onto your devices can use UPNP to communicate the same way a verified secure program would. Since UPNP is a universal protocol, device manufacturers often have their own version of UPNP and there is high risk of having bugs without a standard implementation of the protocol [11].

UPNP may be a high-risk protocol for malicious attacks since it makes a firewall almost useless. Most routers provide a way to disable UPNP for securing a network. However, if UPNP is disabled various device functions may not function properly without correct configuration. UPNP takes care of opening various ports required for torrents, multi-player gaming, and even media connections such as video chat services.

A screenshot of a social media post

Description automatically generated

**Figure 6:** Ports/Hosts tab for 192.168.0.1

The next device we are going to analyze is a laptop running windows 10 and has an IPv4 address of 192.168.0.2. As you can see in **Figure 7** under the “Host Details” tabthis device currently has 5 open ports and no filtered ports. As mentioned previously the OS of the device is Windows 10 build #1067. This device where the intense scan we are analyzing was initiated. However, if you notice there is no information regarding the system uptime, last boot time, IPv6 address or MAC address. This is common when conducting network scans. Due to the many variations in operating systems not all will communicate with NMAP the same way. Some will provide more information to NMAP while others will ignore the incoming requests from NMAP.

The first service listed on host 192.168.0.2 in **Figure 8** is MSRPC running on open port 135. MSRPC the acronym for Microsoft Remote Procedure Call. This service originated in the opensource software pool and was adopted and expanded by Microsoft. The service allows a client to communicate with an application on a remote server. The client uses name resolution to locate the application it needs to communicate with on the remote server [12]. Once the application on a remote server is located the client then asks for permission to access the application. If the remote server authenticates the client, the server will be able to run processes on the remote server through a specified application [12]. This service is useful as it allows the client to farm out processing to other possibly more powerful servers. The client then receives the output from the application on the remote server and continues with its intended use.

A screenshot of a cell phone

Description automatically generated

**Figure 7:** NMAP Host details tab of host 192.168.0.2

A screenshot of a cell phone

Description automatically generated

**Figure 8:** NMAP ports/hosts tab 192.168.0.2

Next we can see in **Figure 8** there is a service called NETBIOS over TCP/IP running on open port 139. NETBIOS uses three distinct channels for communications. The first runs on default port 139 labeled the session channel which enables connections to be established between hosts. The second channel runs on default port 138 and is labeled the datagram channel and is used to conduct data transmissions between hosts. The third channel runs on default port 137 and provides a namespace directory for devices [13]. When users log on or applications start their resource name is dynamically added to the namespace [13]. Like, DNS where an IP address is mapped to a specified human readable name for a device on the network.

The final services running on ports 5800 and 5900 are interconnected and are named VNC-HTTP and VNC respectively. VNC is an acronym for Virtual Network Computing which allows the user to control a host through a browser. VNC is an alternative for the CLI on remote servers if the remote host supports graphics. The application is useful for communicating with Raspberry PI devices that host desktop operating systems. Other uses for this service include remote technical support that allows technicians to gain remote access with graphics from a remote location. Some variations of VNC also offer File Transfer Protocols built into the software.

The third and final host we are going to look at has IP address 192.168.0.21 and is a wifi enabled printer on our sample network. The manufacturer is Brother/HP and as we can see in **Figure 9** is running an operating system called VxWorks. VxWorks is a real-time operating system (RTOS) that is commonly used for embedded systems such as a printer or even automotive systems [14]. VxWorks has some similarities to UNIX systems and has the basic shell, memory management, and support for multiprocessing [14]. You can see that NMAP was able to fully qualify the OS of the host along with uptime, boot time, and MAC address since the host is running a UNIX based system. We can see the host currently has 7 open ports and its current state is available.

A screenshot of a cell phone

Description automatically generated

**Figure 9:** NMAP Host Details tab of 192.168.0.21

You can see in **Figure 10** there is an overlap of services and protocols between 192.168.0.1 and 192.168.0.21. Ports 21, 23, 80, 443 are running ftp, telnet, http, https respectively on their default port numbers. We will not discuss these services since they are repetitive and can be researched by reading the analysis of the host at IPv4 address 192.168.0.1. The first new service running on the printer at IP 192.168.0.21 is named ipp which is short for Internet Printing Protocol running on port number 631. The IPP service is a listener that allows print client systems with a means of communicating with a print service running on a printer [15]. The IPP listening service works in multiple phases, the first requires an IPP request to be sent from the client side the the server IPP is listening on. Upon receiving the request the request is then converted into a context that a print service can understand. The print service then responds to the request by printing the requested documents and returns an aknowlegment and success signal back to the client [15]. The last service we will discuss in **Figure 10** is “jetdirect” which is another print service that is developed by HP. The basic protocol structue of “jetdirect” functions the same as IPP however it uses some proprietary functions native to Brother/HP software [15].

A screenshot of a cell phone

Description automatically generated

**Figure 10:** NMAP Ports/Hosts tab for 192.168.0.21

**Conclusion:**

This NMAP scan has only searched the 1000 most common ports used by services. There may be other open or filtered ports that are not represented by NMAP’s predefined “Intense Scan.” With that in mind you may consider scanning your entire network for both TCP and UDP ports ranging between 1 and 65535. NMAP has an option to conduct this type of scan by using the -p flag and placing a range of 1 – 65535 along with the flag -sU to conduct a full UDP scan on your network as well. Note that by adding the -p and -sU flag to your scans this will greatly increase the amount of time the scan will run for. You may consider running this type of scan once a month or for creating a fully comprehensive report of your network. However, if you are running a standard network health check multiple times a day or week and want to ensure there are no immediate threats on your commonly used communications the “Intense Scan” will provide the functionality necessary.

Moving forward it is important to pay close attention to the services running on open ports. When a port is currently at an open status it is most vulnerable for attacks. Be sure to monitor which services are running on open ports and fully understand how the service works along with why the service is running on an open port. Create firewall rules when necessary for specific ports and services and turn open ports off if the service is not needed to be on an open port. Following these simple rules will help you secure your network now and help deter attackers from targeting your system in the future.

**Bibliography**

[1] Ubuntu. 2019. Retrieved From <https://askubuntu.com/questions/247625/what-is-the-loopback-device-and-how-do-i-use-it#:~:text=The%20loopback%20device%20is%20a,running%20on%20the%20local%20machine.>

[2] Oracle. 2010. Retrieved From <https://docs.oracle.com/cd/E19253-01/816-4554/6maoq01tv/index.html>

[3] Andrew Anderson. 1996. All About Ifconfig. Retrieved From <https://www.tldp.org/LDP/nag/node73.html>

[4] NMAP.org. Retrieved From <https://nmap.org/book/man.html>

[5] Andrew J. Bennieston. 2003. NMAP – A Stealth Port Scanner. Retrieved From <https://nmap.org/bennieston-tutorial/#:~:text=The%20two%20basic%20scan%20types,are%20explained%20in%20detail%20below.>

[6] Aman L. 2017. How Does SSH Work. Retrieved From <https://www.hostinger.com/tutorials/ssh-tutorial-how-does-ssh-work#:~:text=SSH%2C%20or%20Secure%20Shell%2C%20is,remote%20servers%20over%20the%20Internet.>

[7] Geek-University.com. 2019. Retrieved From <https://geek-university.com/ccna/telnet-protocol/>

[8] Oracle. 2010. Retrieved From <https://docs.oracle.com/cd/E19182-01/821-0540/ug_restbd-methods_c/index.html>

[9] letsencrypt.org. 2019. Retrieved From <https://letsencrypt.org/docs/allow-port-80/>

[10] Dah Ming Chiu. 1999. TRAM: A Tree-based Reliable Multicast Protocol. Retrieved From <https://www.researchgate.net/publication/2417645_TRAM_A_Tree-based_Reliable_Multicast_Protocol>

[11] geeksforgeeks.org. 2019. Retrieved From <https://www.geeksforgeeks.org/universal-plug-and-play-upnp/>

[12] Ned Pyle. 2012. RPC over IT/Pro. Retrieved From <https://docs.microsoft.com/en-us/archive/blogs/askds/rpc-over-itpro>

[13] Microsoft. 2012. NetBIOS over TCP/IP. Retrieved From <https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc940063(v=technet.10)?redirectedfrom=MSDN>

[14] TechTarget. 2020. Retrieved From <https://searchnetworking.techtarget.com/definition/VxWorks>

[15] pcmag.com. 2020. Retrieved From <https://www.pcmag.com/encyclopedia/term/printing-protocol>

**Reflection:**

Writing this paper has helped me solidify my understanding of how TCP/IP works. It clarified the fact that all these other services run on top of TCP or UDP. I now feel comfortable explaining what a network mask is, how it works, and why there are different classes of netmasks. Previously I had taken an online networking course that fell short of solidifying my understanding of general networking concepts such as how and why CIDR is useful for subnetting. Another solidifying understanding was researching how SSH communication was established and what was happening in the background when using SSH keys. Lastly, I feel as though I can fully understand the output of the NMAP tool and can use it to identify possible threats to my LAN. I will be doing some more research regarding UPNP and if it will cause a major issue for my significant other if I disable the service from my router. I have been trying to think of a way I could secure my laptop to drop all UPNP communications while leaving the protocol enabled on my router. There was one protocol I could not find good documentation on during this assignment and I would like to know more why there is minimal documentation or how I should correct my search to find more documentation. That service was “printer” running on open port number 515.