Network Security DSV/SU 2021

Laboration 1 (vs 1.2)
2021-02-22

Course Responsible: Yuhong Li
Lab Responsible: Stefan Axelsson

This memo; Stefan Axelsson (with quoted text from Ulf Noring).

Introduction

As we're not able to be in the physical lab this year we'll have to make do with what we have available.

This means that you'll perform the labs on your own computers at home. So you'll need access to a PC running Windows or Linux, or a Mac running MacOS. In addition you need the free tool Wireshark (i.e. just download and install it).

The computer also needs a network connection of some sort that allows you to access the internet.

I've tested these labs on a PC running Ubuntu 20.04 and Windows 10 (both with wired Ethernet and WiFi) but they should run on the other systems as well. If they don't then first look at variations in command line parameters between the different systems (i.e. read the documentation). Or you could install Ubuntu, or install/run it as a virtual machine under a free virtualisation environment, i.e. virtualbox.

Note that this lab is based on the memo for the previous version of the lab, i.e. the lab that takes place in our (physical) security lab. I recommend you also read that memo for the self study questions that are not repeated here.

Background

"This lab focuses mainly on the basics of computer networking that you need to be familiar with before delving into the more complex aspects of network security. You are required to follow the instructions given within this tutorial in a sequential manner without skipping any steps in order for you to benefit from the entire exercise. Answer the questions that have been posed, either in this document or, where required, by explaining the answer to the lab assistants. The self-assessment questions are not to be submitted as answers, but serve as a means for you to gauge whether you have understood the major points raised in the different sections. Make sure you always have the latest version of the assignment instructions. The version number can be found on the cover page of the instructions.

You are expected to have attended the lectures prior to doing the lab exercises, and done some precursory studying of the topics at hand before attempting to do the lab exercises. For this particular lab exercise, you are urged to read widely beyond the course text, and seek knowledge particularly from Wikipedia, or the Internet at large – an effective and efficient search engine will be your friend

during this assignment. The material is readily available out there, in various forms. If you still stumble upon serious problems, after having performed an exhaustive preliminary search for knowledge, you may seek clarification from the course staff. Also, please DO NOT copy-paste commands from this file directly into your terminal window. If you do so, it will cause the issued command to not work properly because of the different symbol formats between Linux and Windows systems."

"Please hand your report as a single PDF document. Preferably include the questions as well as the answers in order to speed up marking of the reports. Documents handed in late, in the wrong format, without names, or not using a standardised reference system will not be marked and will require a resubmission. Please note that your hand-ins will be run through a plagiarism checking tool. Make sure that you properly cite everything that is not your own work. Accepted referencing styles are for example IEEE, Oxford, Harvard, and Vancouver."

"Please do not include any screenshots unless absolutely necessary or asked for in the instructions."

"As this is the first lab, it is meant to be a primer into the world of computer networking, giving you a feel of most of the fundamental precepts that underlie a networked environment. At the end of this lab exercise you should be able to:

Describe the different layers of the OSI model showing an understanding of the various important communication protocols that lie within each layer

- Describe in detail various terminologies used in networking
- Understand the importance and use of Layer 2 and Layer 3 protocols in the
- TCP/IP stack, including the addressing schemes used in either layer.
- Understand the differences between switching and routing.
- Understand subnetting at an introductory level and be able to perform simple subnetting
- Understand the need for transport layer protocols as well as the differences between TCP and UDP
- Understand the concept of ports and how they are used by applications/application layer protocols.
- Understand the uses of core networking protocols such as DHCP and DNS
- Identify artifacts of information within network protocols that could help in understanding deviations from security policies."

Ethernet frame forwarding

"Each network interface present on a system is characterized by a unique identifier called the MAC (Media Access Control) address that commonly consists by 48 bits. The MAC address is usually represented in hexadecimal form such as 01:23:45:67:89:ab."

Question 1

On your computer, list the ethernet (or wireless) adapters. How many are there? What are their MAC addresses?

Based on the MAC addresses can you identify the vendor of the network cards?

Question 2

In which layer of the OSI model does Ethernet switches typically operate?

”These devices are also commonly referred to as the Access Layer of a network since they allow clients (anything with networking capabilities e.g. PCs, gaming consoles, network printers etc.) to connect to the network. Wireless access points (WAP) also operate at the same layer since they allow connectivity to wireless clients as well (e.g. mobile phones, tablet PCs etc.). The main function of such network devices is to forward Ethernet frames based on their MAC addresses, a process called forwarding, as well as to verify the integrity of the transmitted data by validating the CRC value of the received frame. Switches, unlike hubs, do not blindly broadcast every received frame to all other ports. They forward frames only to the particular port that the destination MAC address of the frame indicates, except of course in the case of specially designated broadcast frames.

In order to forward a frame to the correct port, the switch maintains an internal table that maps its physical ports to the MAC address of the devices that are connected to. This table is commonly called content addressable memory (CAM), MAC Address Table or source address table (SAT). The first time the switch receives a packet from a physical port, it adds an entry in its SAT table mapping the port number with the source MAC address of the received frame.”

Question 3

Bring up your interface statistics, how many frames have been sent?

Question 4

Can you find out which other MAC addresses your computer has communicated with in the recent past? (Hint: ARP-cache)

Now it’s time to see some ethernet frames in action.

Start a wireshark capture and quickly start a ’ping’ command[1] to www.su.se (i.e. ”ping www.su.se”). Stop the command after a few packets have been sent (on windows this is automatic). Your ping command should list the ip-address that you pinged. Put in a display filter in Wireshark to only view the ip-address in the ping command (ip.addr==xxx.xxx.xxx.xxx). This means that only traffic to or from, that ip-address is displayed.

Question 5

View the ethernet frame part of the sent/received packets. How many frames are listed? Can you correlate them with the sent ping ip packets? What is the receiving MAC address? Is that the MAC address of the www.su.se server? (If not, what kind of device is it, and who manufactured it?)

Verify that the frames are sent with the MAC address of your computer’s active interface (i.e. that the sending MAC address is the same as the interface you think you’re using).

1 See the next section for a more detailed discussion of the ’ping’ command

Network layer packet forwarding

”In order for the two systems to be able to communicate using the IP protocol at the Network Layer (L3), they must each have a distinct IP address. The addresses must be part of the same subnet. In the absence of auto-configuration networking services, the systems have to be manually assigned a static IP address.”

One common utility to troubleshoot network connectivity issues is to use the ‘ping’ command. You can find helpful documentation about the ping command on Linux by typing ‘man ping’ in the terminal window as well as ‘ping /?’ on Windows. An example ping command in Linux could be:

$ sudo ping -l 51020 -f -s 51020 -a 127.0.0.1

Question 6

What does the above (sudo ping -l 51020 -f -s 51020 -a 127.0.0.1) ping command do?

Which network protocol does the ping utility use to craft the ping request and response packets? On which layer of the TCP/IP stack does this network protocol belong?

Again ’ping’ www.su.se

Question 7

”What options did you use? How many bytes of data did each request carry? What was the TTL value used, and what does the TTL abbreviation stand for? What was the packet loss rate? What was the average round trip time (RTT)?”

Now capture a few packets of a ping to a server that you have not communicated with before (in this session). You can for example chose www.chalmers.se

Using Wireshark: ”Examine the captured traffic coming out from the ‘LinuxClient’ in Wireshark and locate and highlight the first captured frame that carried a ping request. Wireshark displays the actual bytes that consisted this packet in the ‘Packet Bytes’ pane while in the ‘Packet Details’ pane, the bytes are parsed, analyzed and reconstructed in the corresponding layers and protocols. Select and highlight a packet representing a ping request.”

Question 8

”Which protocols are employed for sending a ping request from one system to another? Can you describe the order of protocol encapsulation applied according to the TCP/IP or OSI model? What are the values for the type and code fields of the ping request? What is the ID value of the IP packet? What is the type of the Ethernet frame?”

”Highlight a ping reply packet below the previously selected frame. Examine the values of all protocol fields and compare them with those of a ping request packet. Which fields have the same value between them?”

Notice that just before the ping requests and replies were exchanged, two frames using the ARP protocol ***MAY*** have been captured. If you cannot find any ARP frames regarding the server then see if you find other ARP frames in your capture. You may have to wait a long time (minutes) depending on the type of network and how busy it is. (If you cannot then just answer the following questions based on your understanding).

Question 9

What does ARP stand for? In which layer of the TCP/IP stack does it belong? Briefly describe its main function. Briefly describe and explain why this protocol is important in a switched network.

Examine the ARP request packet. What is the destination MAC address? Briefly explain this.

Examine the ARP reply packet. What is the source MAC address of the frame and to which network host does it belong? Does the ARP reply contain valid data?

Note that modern operating systems maintain an internal data structure, commonly named the ARP cache, where previously received ARP replies are stored and examined before any future communication. In both windows and linux, you can type ‘arp -a’ in the command line to examine the current entries in the list. If you cannot capture an ARP packet, then check for entries in the ARP cache, and then try and wait for one of those devices to turn up in a capture. (This may take a minutes or two).

Transport layer

”Up to here, you have managed to create a simple network between two systems and have verified that they can indeed communicate. Building on this ability, the next step is to be able to provide some form of networked application service on one system that the other may need to access and interact with. This has given rise to the server-client paradigm. One system provides specific services over the network (e.g. a web server, email server, file server or authentication server) which other systems can access. One physical host can simultaneously act as a server for more than one network services while also being a client of services offered by others. Commonly, servicing or consuming network services translates to application processes running on the system (either by the user or the OS) with the function of translating user preferences or UI actions into messages that are encapsulated and transmitted via network packets.

In order for the communication to succeed however a formalized manner of message exchange has to be agreed upon and implemented, thus allowing communication between systems of different OS or configurations. Formats and rules that are specified and followed by the communicating host comprise a network protocol. Network protocols in the transport layer deal with establishing a logical host-host communication. Network protocols in the application layer build atop the transport layer by establishing a logical process-to-process communication. This allows multiple processes between two

systems to exchange messages simultaneously. The ability to multiplex and de-multiplex multiple processes' streams is provided through the concept of network ports. Processes are bound to network ports that mark their corresponding data streams with a number identifier. A port is associated with the combination of the IP address of a system's network interface along with a transport layer protocol such as TCP or UDP and is identified by a 16-bit number."

We'll be becoming acquainted with the 'nc' i.e. netcat command. This is a "Swiss army knife" network tool that even though basic can be very useful when troubleshooting networks (or even implement simple network services). Now, it of course is usually used to communicate between two different computers, but as we only have access to one computer, we'll simulate server and client on the same computer. (If you have two or more computers available that are connected to the same network then feel free to run netcat on the different computers instead).

Start by opening a port that netcat can listen on. Open a terminal window and give the commands

nc -l 21 and nc -l 30000.

Question 10

Have the commands completed successfully? Why or why not? If a command has failed, how can the command be made to work? What does the 'l' parameter mean? (Note that the success/failure here may depend on the operating system you're using.)

Open two command shell windows and type 'nc -l 30000' in one of them and 'nc 127.0.0.1 30000' in the other. (You may have to use 'nc -l 127.0.0.1 30000' to start the server side connection.)

Also start a Wireshark session, but make sure to capture from the "loop back" interface, not your usual network interface.

Type in one window and confirm that the text shows up in the other window. (Do this for both windows).

Question 11

View the traffic in Wireshark and answer the questions: "Which transport layer protocol has been used for the communication? Provide a screenshot showing that the communication has been properly established. What are the client and server ports? How many packets flow from client to server and how many vice-versa? How many bytes are sent in each direction and in total?" Can you recover the text you sent using Wireshark? What was it?

Question 12

What were the server and client ports that were used in this communication session?

Question 13

Also, what does 127.0.0.1 mean in this context? What is special with this ip address?

Retry the previous experiment but with two active parallell sessions. I.e. open four windows and start two nc instances listening on different ports; 30000 for the first, and 31000 for the second. To make it easy for you to identify them, I suggest you type ”HELLO” in one conversation, and ”WORLD” in the other.

Question 14

Which fields and values of the captured packets give you an idea of which is the client and which is the server in each individual session?

Now we’ll try to add the -u parameter to nc. Open two windows. Type ’nc -l -u 127.0.0.1 30000’ in one and ’nc -u 127.0.0.1 30000’ in the other. Try to communicate between the two instances as you’re watching the traffic in Wireshark.

Question 15

What are the differences from the TCP experiment you did above?

Application Layer

”Besides the actual data that application layer protocols such as HTTP, FTP, SSH, DNS etc. may exchange, they also specify their own distinctive set of commands and format of messages that they use to serve their own specific role. Protocol analyzers such as Wireshark support a large variety of such protocols and can interpret and present all these additional metadata in a friendly manner.”

The next step is to communicate with a web server. Start Wireshark and a web browser. Surf to http://neverssl.com

Examine the captured packets that were generated by this request.

Question 16

Which application layer protocol is used? Can you identify a packet that carries the initial web request? Write down which metadata fields appear in the request and what their value means?

Can you identify the packet that carries the response? Can you identify any interesting metadata fields? Can you extract the transmitted HTML code of the visited web page? If not why not? (Note that there is no encryption: *neverssl* is called that for a reason).

IP Addressing and subnets

For two systems to be able to communicate successfully over a network, they must be identified by distinctive IP addresses. However, IPv4 addresses are not unlimited since an Ipv4 address consists of 32 bits (amounting to 2 32 , or approximately a little more than 4 billion IPs). IP address ranges are managed by international organizations and are further split and provided to companies, universities, ISPs etc.

Question 17

What is the lowest and the highest IP address that belong to the IP address range of the adress of your interface? What is the broadcast address of this IP address range? How many hosts can your network support?

Question 18

Write down the IP address that you have been assigned. Identify the network ID of this IP address as well as the host ID of it. A technique called subnetting allows breaking a given IP address range into smaller, better manageable blocks. A subnet mask groups the network prefix along with some high-order bits from the host part into forming an ‘extended’ network prefix. The subnet mask is used to determine the Network ID of an IP address via the logical AND operation in the process known as “Binary AND–ing”.

Question 19

Splitting the above IP address range in half, gives you two smaller subnets, each with a subnet mask of /25 (i.e. 255.255.255.128). What is the IP address range and the broadcast address of each subnet? How many hosts can they support each? Pick one IP adress belonging to each subnet and write them down along with their network ID and host ID.

”In order for a host to send data to a destination host in the same network, it has to resolve the latter’s MAC address based on the destination IP address using ARP. However, before even trying to do that, the host must examine if the destination system belongs to the same network or not. This process is accomplished by consulting a local host routing table that each host maintains. In order to view the routing table of a Windows 7 system type ‘route print’ in the command line. For a Linux system you can type ‘route’ in a terminal window.”

”The OS processes this table in a bottom to top manner. Whenever any application or process instructs the OS to send network data to a destination host, the destination IP address is ‘AND’ed with the netmask value of each entry successively. If the resulting network ID matches with the value present in the first column, then the packet can be forwarded to the specified gateway. ‘On-link’ means that the destination is on the same network so the packet can be sent directly through the switch given that the destination MAC address is known or discovered.”

Examine the routing table of your host.

Question 20

What is the IP address range of systems with which your system can communicate? Is it a private or a public IP-address? What’s the difference between these two types of addresses and what are the advantages and disadvantages of private IP-addresses.

Basic Internetworking

As seen before, it should be clear by now that nodes belonging to the same network communicate directly via the switch using the source and destination IP and MAC addresses. Since ARP provides a mapping between the logical IP address and the physical MAC address, the forwarding decisions are actually based only on Layer 2. In order for two systems belonging to different networks to be able to communicate, a routing process between the two networks must be applied. Routers are dedicated network devices with multiple network interfaces that connect to disparate networks and enable forwarding of IP packets from one network to another.

However, since we're not in the lab it will be difficult to do actual routing.

Question 21

Instead list the topology of your network, identifying your host and the (typically only one) router that it attached to.

Question 22

What is the IP-address of the router interface you communicate with?

Start Wireshark and send a ping packet to a host on the internet. Do it for long enough to invalidate the ARP-cache of your host, and capture the ARP request for your router interface.

Question 23

Study the capture. Note the recipient MAC-address of the ARP packet you are sending to the network. What address is that? What's special about it? Why does it work this way? (i.e. how does your computer find the address of a host that can answer ARP requests? It can't of course use an IP-address, as it doesn't know that yet…) You may have to try a few times to get a MAC-address that is not specific to a computer...

Question 24

Describe briefly what the difference is between a host that receives an IP-packet with a destination address that is not its own, and a what a router does when it receives an IP-packet on an interface where that address does not belong to the router. (As is the case here). Hint: look up the term "forwarding".

Application Layer Protocols (DHCP and DNS)

"There are hundreds of different application layer protocols, all providing different types of network services and applications. Some of these application layer protocols provide important services to the network itself and its users They are commonly used in most modern networks. DNS (Domain Name System) is a network protocol that provides a naming service mapping IP addresses to more human-friendly names (e.g. www.google.com). The DNS infrastructure is based on hierarchically distributed databases starting from the root zones (the actual '.'), spanning down to top level domains (TLDs) or country- code top level domains (e.g. '.com', '.org', '.se', '.uk') which in their turn span further to subdomains (e.g. '.example.com', '.su.se') and recursively into subdomains of them (e.g '.dsv.su.se'). Domain names that are associated with IP addresses comprise hostnames (e.g. 'www.google.com,

cs2lab.dsv.su.se'). A client accesses a DNS server in order to resolve a hostname into its respective IP address. A DNS server that is not aware of an answer to a specific DNS query may either consult hierarchically upper DNS servers (recursive query resolution) or refer the client to possibly more authoritative DNS servers (iterative query resolution)."

Question 25

Find out which DNS server you're currently configured to use. Where is that server (i.e. who runs that server). Do you have one or two (or several) DNS-servers configured? Why may there be more than one?

Now start Wireshark to capture the DNS-traffic and access a web-server you have not accessed in quite some time (preferably never; just google something obscure and chose a site at random).

Question 26

Check the captured DNS-traffic: "What port does the DNS server use for the communication? Which transport layer protocol is used? Which flag(s) did the DNS query header have enabled and what is the Transaction ID? Of which type is the query?"

"The received DNS reply is cached locally by the OS for as much time as the TTL field indicates. You can clear the local DNS cache on Windows by issuing the command 'ipconfig /flushdns' in an administrative command line."

DHCP

"DHCP (Dynamic Host Configuration Protocol) automates the process of configuring a network device so as to be able to communicate with other devices in the network. All these settings that used to be manually configured (IP address, subnet mask, default gateway, DNS server) can be provided automatically to any newly connected device to the network by the DHCP server."

As you're probably on a DHCP enabled network and only have one computer it's difficult to catch the initial DHCP query response, but if you're on a network with other devices you may see one, especially if you can turn off another device and turn it on again.

Question 27

Do that and see if you can capture the DHCP request/response (if not just say that it wasn't possible). Study the DHCP request; how does the computer know which host can give it an IP-address *before* it can communicate on the IP-network (as it doesn't yet have an address)?

That concludes lab 1