UNIVERSITY OF WESTMINSTER

Network Security Fundamentals and Threats

6COSC019W- Cyber Security

Dr Ayman El Hajjar

January 30, 2023

School of Computer Science and Engineering
University of Westminster

OUTLINE

1. Networks fundamentals
2. Network Malicious Attacks
    - Application layer attacks
        - HTTP
        - DNS
        - Email protocols
    - Transport layer protocols
    - Session layer attacks
    - Transport layer attacks
    - Network layer attacks
    - Data Link layer attacks

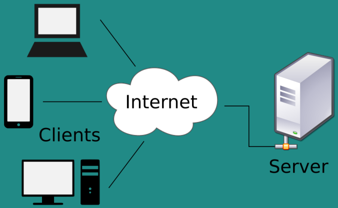
Networks fundamentals

INTRODUCTION TO NETWORKING

- ✺ What is a Network?
  - ☞ Set of technologies that connects computers
  - ☞ Allows communication and collaboration between users
  - ☞ Collection of computers and devices connected together
- ✺ The uses of a network
  - ☞ Simultaneous access to data
  - ☞ Shared Resources
  - ☞ Personal communication
  - ☞ Easier data backup
- ✺ Main types of Networks
  - ☞ Wide Area Network (WAN): Connect systems over a large geographic area
  - ☞ Local Area Network (LAN) : Provide network connectivity for computers located in the same geographic area

Network architecture: Centralized Architecture

- Client/server architecture is a **centralized architecture**.
- A client/Server architecture exist is when one main device in order to get a service. In this instance, a server is the device providing a service
- All devices connected to it are clients.
- Most of the internet is of Client/Server architecture.
- A website is an example of a client/server architecture



NETWORK ARCHITECTURE: DE-CENTRALIZED ARCHITECTURE

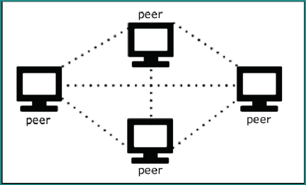
✺ Peer to peer architecture (Ad Hoc): Devices are connected directly to each other.

✺ This is a **decentralized architecture**. There is no centralized entitiy that controls the communication.

✺ Each device can be server or client depending on whether it is sending or receiving

✺Examples of peer to peer networks:

- ☞ Bluetooth
- ☞ NFC



De-Centralized Architecture: Peer to Peer

The Internet

What is the Internet

The simplest definition of the Internet is that it's a network of computer networks

OSI Security Architecture

- Security attack
  - Any action that compromises the security of information owned by an organization
- Security mechanism
  - A process (or a device incorporating such a process) that is designed to detect, prevent, or recover from a security attack
- Security service
  - A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization
  - Intended to counter security attacks, and they make use of one or more security mechanisms to provide the service

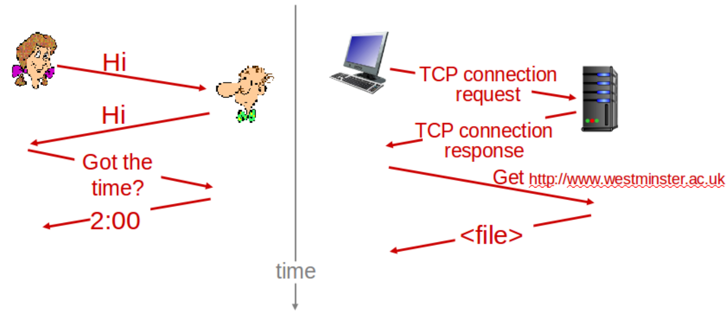
WHAT IS A PROTOCOL?

- A protocol is a set of rules or conventions that dictate communication.
    - Without communication protocols, I might ask a question and you might respond by closing your eyes.
    - Communication in this instance did not actually happen, or it was meaningless.
    - In another word, the communication failed because it did not follow the rule.
    - The rule is- If and when we ask a question, we expect a response.
        - Regardless of what the response is!

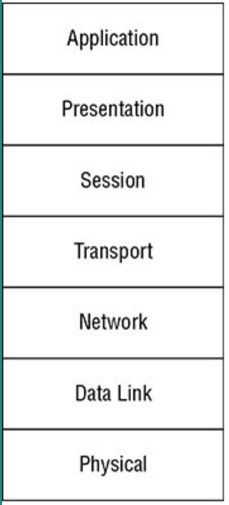
What is a protocol?

Protocol

- ✺ A protocol is a set of rules and formats that govern the communication between communicating peers.
  - ☞ set of valid messages
  - ☞ meaning of each message
- ✺ A human protocol and a computer network protocol



OPEN SYSTEM INTERCONNECTION (OSI) MODEL



Application

Presentation

Session

Transport

Network

Data Link

Physical

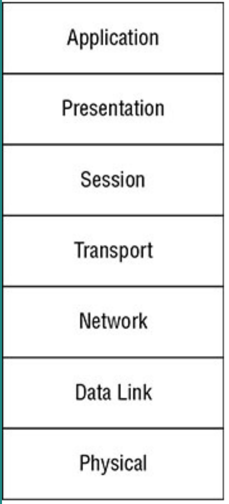
The OSI model consists of seven separate and distinct layers, each describing a particular set of functions and behaviors.

Although every protocol used for communication will fit into one of these seven layers, not all communication streams will make use of all seven layers.

Order is essential- Remember:

Please **D**o Not **T**ell **S**ecret **P**asswords **A**nymore

OSI LAYERS MODEL



Application

Presentation

Session

Transport

Network

Data Link

Physical

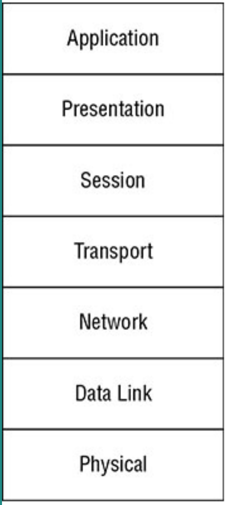
✺ Application layer

- ☞ The Application layer is the one closest to the user
- ☞ Application layer protocols manage the communication needs of the application.
- ☞ They may identify resources and manage interacting with those resources.

✺ Presentation layer

- ☞ The Presentation layer is responsible for preparing data for the Application layer.
- ☞ It makes sure that the data that is handed up to the application is in the right format so it can be consumed.
- ☞ When systems are communicating, there may be disconnects in formatting between the two endpoints and the Presentation layer makes sure that data is formatted correctly ie. Unicode, or ACSII code.

OSI LAYERS MODEL



Application

Presentation

Session

Transport

Network

Data Link

Physical

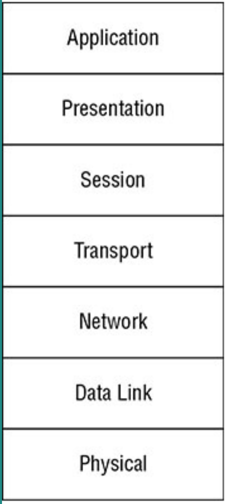
✺ Session layer

- ☞ he Session layer manages the communication between the endpoints when it comes to maintaining the communication of the applications (the client or server).
- ☞ Remote procedure calls (RPCs) and file sharing are examples of functions at the Session layer.

✺ Transport layer

- ☞ The Transport layer takes care of segmenting messages for transmission.
- ☞ Both the TCP and the UDP are transport protocols. These protocols use ports for addressing, so receiving systems know which application to pass the traffic to.

OSI LAYERS MODEL



✺ Network layer

☞ The Network layer gets messages from one endpoint to another.

☞ It takes care of addressing and routing.

☞ The IP is one protocol that exists at this layer.

✺ Data Link layer

☞ Media Access Control M(MAC) address is a layer 2 addressing. It identifies the network interface on the network so communications can get from one system to another on the local network.

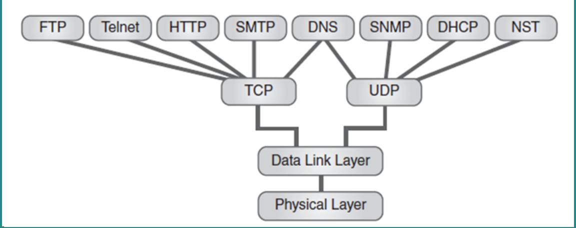
☞ They take care of formatting the data to be sent out on the transmission medium.

✺ Physical layer

This layer probably speaks for itself. This is all the protocols that manage the physical communications.

TCP/IP AND HOW IT WORKS

- A suite of protocols that operate at both the Network and Transport layers of the OSI Reference Model
- Governs all activity across the Internet and through most corporate and home networks
- Developed by the DoD to provide a highly reliable and fault-tolerant network infrastructure (security was not a focus)

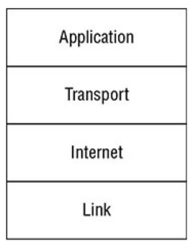


FOUR LAYERS OF THE TCP/IP MODEL

OSI had to be abstract and flexible in order to accommodate a wide variety of protocols and designs.

TCP/IP, on the other hand, as an as-built definition, is only four layers.

The TCP/IP architecture is a much simpler design than the OSI model.



- Session, Presentation, and Application layers from the OSI model are collapsed into the **Application layer**.
- **Transport layer** is the same in both models.
- The **Internet** and Network layers are named and function very similarly
- Physical and Data Link layers from the OSI model are collapsed into the **Link layer** in the TCP/IP model

Network Malicious Attacks

MALICIOUS ACTIVITY ON THE RISE

- Examples of the malicious attacks are everywhere
- Data breaches occur in both public and private sectors
- In 2020, China was top country of origin for cyberattacks, at 41 percent.
- United States was second at 10 percent.
- Real time attacks maps below:
  - ☞ DDoS real time attacks
  - ☞ Cyberthreats real time map
  - ☞ Cyberthreats real time map

PASSIVE ATTACKS- AN EXAMPLE (HTTP)

✺ Networks and computers are potentially subject to all kinds of attack.

✺ A Passive attack attempts to learn or make use of information from the system but does not affect system resources.

✺ Passive Attacks are in the nature of eavesdropping on or monitoring of transmission. The goal of the opponent is to obtain information is being transmitted.

✺ Example of such attacks are:

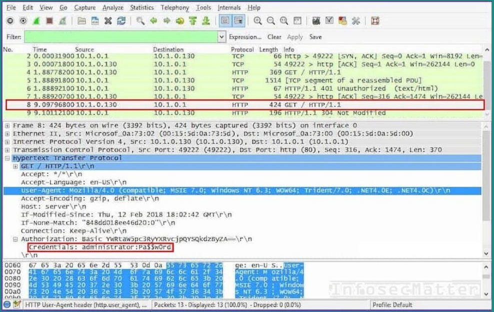
- ☞ “snooping” or “listening” Information are sent over a network as plain text

✺ Anyone listening to the traffic can hear all this “noise” by setting their NIC into promiscuous mode (or monitor mode)

- ☞ This is called packet sniffing
- ☞ i.e. accepting all packets, even those not meant for them

HTTP BASIC AUTHENTICATION- PLAIN TEXT

- ✺ The most basic authentication. Authentication is simply based on the existence of an IP address or not.
- ✺ Browser cache the credentials for a period of time (a form of session).
- ✺ Insecure as full credentials pass over the wire and
- ✺ Data sent in the clear



```
No.   Time       Source      Destination  Protocol Length Info
    2 0.00031900 10.1.0.1    10.1.0.130   TCP        66 http > 49222 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=
    3 0.00071800 10.1.0.130  10.1.0.1     TCP        54 49222 > http [ACK] Seq=1 Ack=1 Win=262144 Len=0
    4 1.88778200 10.1.0.130  10.1.0.1     HTTP      369 GET / HTTP/1.1
    5 1.88891800 10.1.0.1    10.1.0.130   TCP      1514 [TCP segment of a reassembled PDU]
    6 1.88892100 10.1.0.1    10.1.0.130   HTTP       67 HTTP/1.1 401 Unauthorized  (text/html)
    7 1.88920700 10.1.0.130  10.1.0.1     TCP        54 49222 > http [ACK] Seq=316 Ack=1474 Win=262144 Le
    8 9.09796800 10.1.0.130  10.1.0.1     HTTP      424 GET / HTTP/1.1
    9 9.10112100 10.1.0.1    10.1.0.130   HTTP      196 HTTP/1.1 304 Not Modified
```

```
⊞ Frame 8: 424 bytes on wire (3392 bits), 424 bytes captured (3392 bits) on interface 0
⊞ Ethernet II, Src: Microsof_0a:73:02 (00:15:5d:0a:73:5d), Dst: Microsof_0a:73:00 (00:15:5d:0a:5d:00)
⊞ Internet Protocol Version 4, Src: 10.1.0.130 (10.1.0.130), Dst: 10.1.0.1 (10.1.0.1)
⊞ Transmission Control Protocol, Src Port: 49222 (49222), Dst Port: http (80), Seq: 316, Ack: 1474, Len: 370
⊟ Hypertext Transfer Protocol
  ⊞ GET / HTTP/1.1\r\n
    Accept: */*\r\n
    Accept-Language: en-US\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.3; WOW64; Trident/7.0; .NET4.0E; .NET4.0C)\r\n
    Accept-Encoding: gzip, deflate\r\n
    Host: server\r\n
    If-Modified-Since: Thu, 12 Feb 2018 18:02:42 GMT\r\n
    If-None-Match: "848dd018ee46d20:0"\r\n
    Connection: Keep-Alive\r\n
  ⊟ Authorization: Basic YWRtaW5pc3RyYXRvcjpQYSQkdzByZA==\r\n
      Credentials: administrator:Pa$$w0rd
    \r\n
```

```
0060  67 65 3a 20 65 6e 2d 55  53 0d 0a 55 73 65 72 2d   ge: en-U S..User-
0070  41 67 65 6e 74 3a 20 4d  6f 7a 69 6c 6c 61 2f 34   Agent: M ozilla/4
0080  2e 30 20 28 63 6f 6d 70  61 74 69 62 6c 65 3b 20   .0 (comp atible; 
0090  4d 53 49 45 20 37 2e 30  3b 20 57 69 6e 64 6f 77   MSIE 7.0 ; Window
00a0  73 20 4e 54 20 36 2e 33  3b 20 57 4f 57 36 34 3b   s NT 6.3 ; WOW64;
```

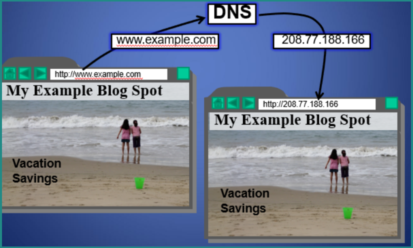
HTTP User-Agent header (http.user_agent), ... Packets: 13 · Displayed: 13 (100.0%) · Dropped: 0 (0.0%) Profile: Default

Domain Name System

✺ **The domain name system (DNS)** is an application-layer protocol for mapping domain names to IP addresses

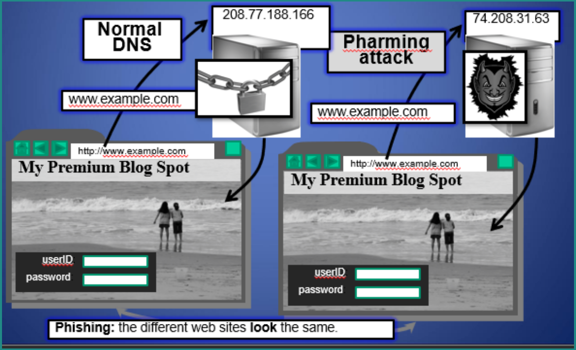
✺ DNS provides a distributed database over the internet that stores various **resource records**, including:

- ☞ Address (A) record: IP address associated with a host name
- ☞ **Mail exchange**(MX) record: mail server of a domain
- ☞ **Name server** (NS) record: authoritative server for a domain



DNS attacks- Pharming: DNS Hijacking

- ✺ An attacker attempt to change the IP associated with a server maliciously:

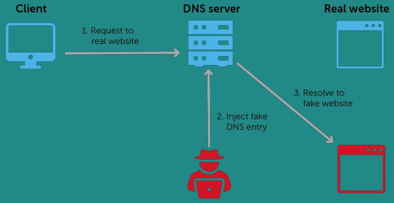


Phishing: the different web sites **look** the same.

DNS attacks- DNS cache poisoning

- ✺ **Basic idea:** give DNS servers false records and get it cached
- ✺ DNS uses a 16-bit request identifier to pair queries with answers
- ✺ Cache may be poisoned when a name server:
  - ☞ Disregards identifiers
  - ☞ Has predictable ids
  - ☞ Accepts unsolicited DNS records

DNS poisoning



DNS Cache Poisoning Prevention

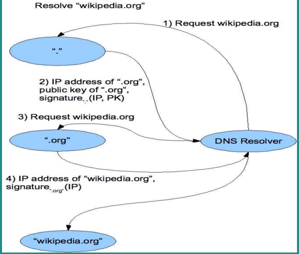
✺ As the internet becomes regarded as critical infrastructure there is a push to secure DNS

✺ May add considerable load to dns servers with packet sizes considerably larger than 512 byte size of UDP packets

✺ Deploy DNS Security (DNSSEC) to ensure:

- ☞ Authenticity of DNS answer origin
- ☞ Integrity of reply
- ☞ Authenticity of denial of existence
- ☞ Accomplishes this by signing DNS replies at each step of the way
- ☞ Uses public-key cryptography to sign responses

DNS Signing



TRANSPORT LAYER: USER DATAGRAM PROTOCOL (UDP)

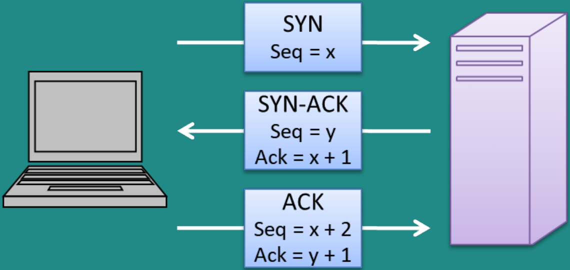
- Lightweight and connectionless
- Small packet sizes (60% less than TCP), in header size UDP (8 bytes) & TCP (20 bytes)
- No connection to create and maintain
- More control over when data is sent
- Does not compensate for loss of packet
- Does not deliver or guarantee packet delivery in order
- Does not check if network is busy

TRANSPORT LAYER: TRANSMISSION CONTROL PROTOCOL (TCP)

- ✺ Reliable and connection-based
  - ☞ Sequence numbers, timeouts, and retransmissions protect against loss and reordering.
  - ☞ Sequence numbers: loss, reordering, duplication.
  - ☞ Timeouts: loss.
  - ☞ Retransmission: loss
- ✺ TCP packets have a header section with a flags field
- ✺ Consider 4 of the possible flags
  - ☞ SYN (Synchronise)
  - ☞ ACK (Acknowledge)
  - ☞ FIN (Finished)
  - ☞ RST (Reset)

TRANSPORT LAYER: TCP PACKETS

- Three way handshake TCP packets exchange
  - ☞ To initiate a TCP connection the initiating system sends a SYN packet to the destination.
  - ☞ Destination sends an ACK to acknowledge the receipt of the first packet (a combined SYN/ACK packet).
  - ☞ The first system sends an ACK packet to acknowledge receipt of the SYN/ACK
  - ☞ Data Transfer can then begin!



SESSION HIJACKING

✺ In a session hijacking attack, an attacker takes control of or modifies any communications between two hosts.

- ☞ Communications can be anything from a Telnet session, or a domain name lookup to a local user's keystrokes.

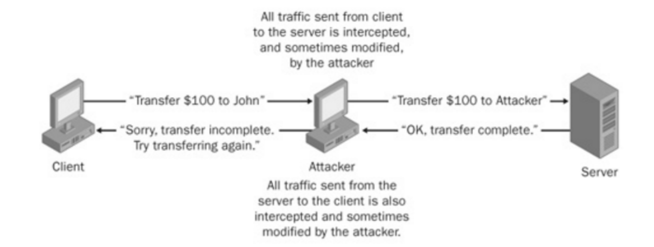
✺ Session hijacking takes advantage of the fact that most communications are protected from the beginning at session setup, such as by providing credentials, but not during the session.

✺ Session hijacking attacks generally fall into the following three categories:

- ☞ Man in the middle attack
- ☞ Blind hijack attacks
- ☞ Session theft attacks

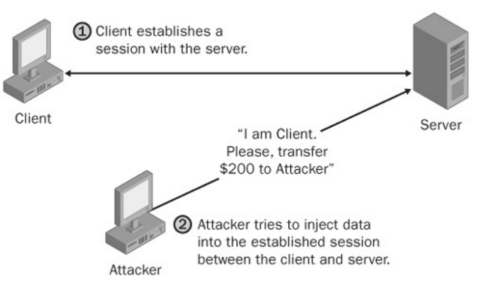
Man in the middle attack (MITM)

✺ An attacker intercepts all communications between two hosts.
✺ The attacker positions themselves so that communications between a client and server must flow through them, which allows them to modify the communications.
✺ Protocols that rely on the exchange of public keys to protect communications, for example, are often the target of these types of attacks (ARP, DNS)


All traffic sent from client to the server is intercepted, and sometimes modified, by the attacker
"Transfer $100 to John"
"Transfer $100 to Attacker"
"Sorry, transfer incomplete. Try transferring again."
"OK, transfer complete."
Client
Attacker
Server
All traffic sent from the server to the client is also intercepted and sometimes modified by the attacker.

BLIND HIJACK ATTACK

- ✹ An attacker can inject data such as malicious commands into those communications
- ✹ This type of attack is called blind hijacking because the attacker can only inject data into the communications stream;
- ✹ The attacker cannot see the response to that data, such as ”The command completed successfully.
- ✹ This method of hijacking is still very effective.


① Client establishes a session with the server.
Client
Server
“I am Client. Please, transfer $200 to Attacker”
② Attacker tries to inject data into the established session between the client and server.
Attacker

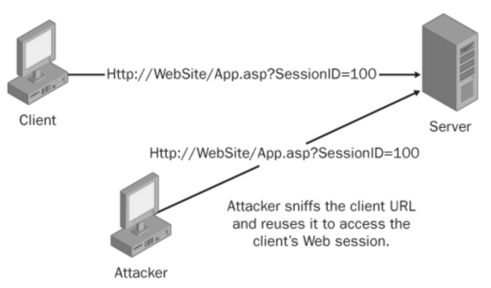
SESSION THEFT ATTACKS

✷ In a session theft attack, the attacker is neither intercepting nor injecting data into existing communications between two hosts.

✷ Instead, the attacker creates new sessions or utilizes old sessions.

✷ Repeat sessions !!

✷ This type of session hijacking attack is most common at the application level, such as a Web application.

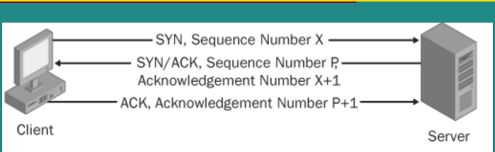

Client
Http://WebSite/App.asp?SessionID=100
Server
Http://WebSite/App.asp?SessionID=100
Attacker

Attacker sniffs the client URL and reuses it to access the client's Web session.

SO HOW DOES SESSION HIJACKING HAPPEN?

- Hijacking Session hijacking at the network level is especially attractive for attackers.
- They don’t need to have access on a host as they do with host-level session hijacking.
- Nor do they need to customize attacks on a per-application basis as they have to at the application level.
- Network-level session hijacking attacks allow attackers to remotely take over sessions, usually undetected
- The two protocols used are TCP and UDP.
  - Hijacking a TCP session
  - Hijacking a UDP session (Out of scope of this module)

HIJACKING A TCP SESSION


SYN, Sequence Number X
SYN/ACK, Sequence Number P,
Acknowledgement Number X+1
ACK, Acknowledgement Number P+1
Client
Server

✺ **Enter the attacker**:

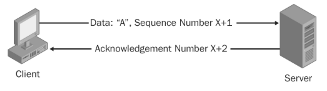
☞ Spoof the client’s IP address: Easy

☞ Determine the correct sequence number the server is expecting from the client. Nothing a good network sniffer can’t figure out.

☞ Inject data into the session before the client sends its next packet.

✺ **Note:** The attacker needs a way to ”hold down” the client from sending into the session new data that would shift sequence numbers forward (DoS) client or send before.

HIJACKING A TCP SESSION



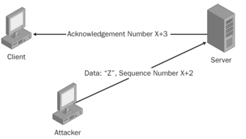
- ✺ **Enter the attacker**:
  - ☞ Spoof the client's IP address: Easy
  - ☞ Determine the correct sequence number the server is expecting from the client. Nothing a good network sniffer can't figure out.
  - ☞ Inject data into the session before the client sends its next packet.

✺ **Note:** The attacker needs a way to "hold down" the client from sending into the session new data that would shift sequence numbers forward (DoS) client or send before.

Hijacking a TCP session

- **Enter the attacker**:
  - Spoof the client's IP address: Easy
  - Determine the correct sequence number the server is expecting from the client. Nothing a good network sniffer can't figure out.
  - Inject data into the session before the client sends its next packet.

- **Note:** The attacker needs a way to "hold down" the client from sending into the session new data that would shift sequence numbers forward (DoS) client or send before.



PORT SCANNING

- Port scanning is an essential step in the reconnaissance phase
- Several scans exist, each reveals different type of information.
  - ☞ **Ping Scan**: The ping scan sends a single ICMP echo request from the source to the destination device. A response from an active device returns an ICMP echo reply, unless the IP address is not available on the network or the ICMP protocol is filtered.
  - ☞ **Connect scan:** Fully connect to the target ip address and port in a complete TCP handshake. Reliable but very noisy.
  - ☞ **Syn Scan:** Sends syn requests to the target to gather information about open ports without completing the TCP handshake. When an open port is identified, the TCP handshake is reset before it can be completed. This technique is sometimes called to as ”half open” scanning.

    Fin Scan: Sends a FIN (or finish) packet to target. If that port is not listening, no response. If it is listening an error response is received.

IP VULNERABILITIES

- Unencrypted transmission
  - ☞ **Eavesdropping** possible at any intermediate host during routing
- No source authentication
  - ☞ Sender can **spoof source address**, making it difficult to trace packet back to attacker
- No integrity checking
  - ☞ Entire packet, header and payload, can be modified while en route to destination, enabling **content forgeries**, **redirections**, and **man-in-the-middle** attacks
- No bandwidth constraints
  - ☞ Large number of packets can be injected into network to launch a **denial-of-service** attack

Denial-of-Service Attacks

Denial-of-Service Attacks

✺ One of the most common types of attacks. It prevents legitimate users from accessing the system

✺ The idea is that computers have physical limitations

- ☞ Number of users
- ☞ Size of files
- ☞ Speed of transmission
- ☞ Amount of data stored

✺ Exceed any of these limits and the computer will cease to respond

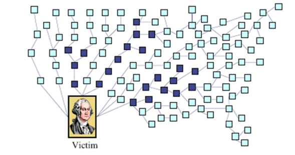
DISTRIBUTED DENIAL OF SERVICE ATTACKS

- ✺ A DoS attack attempts to prevent valid users from accessing network resources.
- ✺ A distributed denial of service (DDoS) attack has the same goal but amplifies the DoS attack by using multiple hosts.
- ✺ Whereas a DoS attack would overwhelm the network connection for a targeted host through a more powerful host, a DDoS attack would use multiple intermediary hosts to generate enough traffic to disrupt server farms or a whole network segment, and possibly beyond.
- ✺ A challenge to detect a DDos is that traffic is coming from several ip addresses. That makes it more difficult to detect until it is too late

Denial of Service Attack

- ✺ Send large number of packets to host providing service
  - ☞ Slows down or crashes host
  - ☞ Often executed by botnet
- ✺ Attack propagation
  - ☞ Starts at zombies
  - ☞ Travels through tree of internet routers rooted
  - ☞ Ends at victim
- ✺ IP source spoofing
  - ☞Hides attacker
  - ☞ Scatters return traffic from victim

DDoS



OTHER DOS ATTACKS

- ✺ The Ping of Death (PoD)
  - ☞ Sending a single large packet.
  - ☞ Most operating systems today avoid this vulnerability.
  - ☞ Still, keep system patched.

Teardrop Attack

  - ☞ Hacker sends a fragmented message
  - ☞ Victim system attempts to reconstruct message
  - ☞ Causes system to halt or crash
- ✺ DHCP Starvation
  - ☞ If enough requests flooded onto thea network, the attacker can completely exhaust the address space allocated by the DHCP servers for an indefinite period of time. This is a DoS attack is called DHCP starvation. There are An attacker can use a tools such as The Gobbler that will do this for the attacker to easily commit this type of attack.

DDOS MIRAI BOT ATTACK

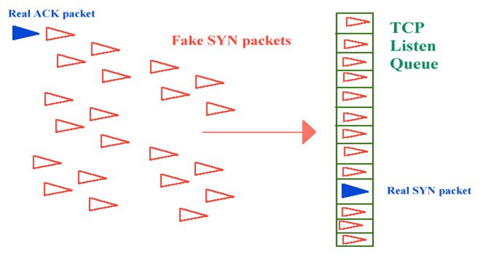
- ✷ Typical DDos attack few years ago generated an average 200Mbps.
- ✷ Mirai Botnet infected cameras (IP CCTV), printer and routers and thousands of other deices.
- ✷ Devices are infected by the Mirai botnet malware.
- ✷ it is reported that 100000 devices participated were zombies
  - ☞ An army of the undead, wreaking havoc on the Internet it's a nightmare scenario that has played out time and again as the world's online population has exploded.
- ✷ Zombies were always present on the Internet- They are as old as malware.

WHAT CHANGED

- Internet Of Things is the main disruptive technology that made such attacks possible.
  - Increase in number of unattended devices (CCTV, smart devices, etc..)
- Mirai botnet attack in 2016 attack generated an average of 1 Terabit per second.
- The result
  - Yahoo, Ebay, Amazon, CNN, ZDNet were out for most of the day of the attack.
- November 2016 till now victims is in the hundreds of thousands of websites
- How to mitigate / protect against such atacks
  - Buy enough bandwidth for your website
  - Too expensive

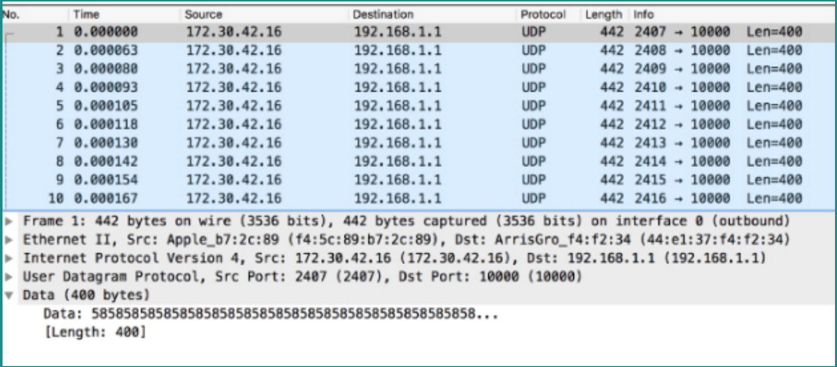
TCP SYN Flood

- TCP SYN Flood Attack (DoS)
  - Hacker sends out a SYN packet.
  - Receiver must hold space in buffer.
  - Bogus SYNs overflow buffer.



UDP FLOOD

- ✷ UDP Flood Attack (DoS)
  - Hacker sends UDP packets to a random port
  - ☞ ☞ Generates illegitimate UDP packets
  - ☞ Causes system to tie up resources sending back packets

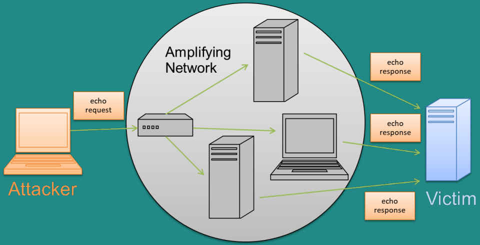


No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.30.42.16	192.168.1.1	UDP	442	2407 → 10000 Len=400
2	0.000063	172.30.42.16	192.168.1.1	UDP	442	2408 → 10000 Len=400
3	0.000080	172.30.42.16	192.168.1.1	UDP	442	2409 → 10000 Len=400
4	0.000093	172.30.42.16	192.168.1.1	UDP	442	2410 → 10000 Len=400
5	0.000105	172.30.42.16	192.168.1.1	UDP	442	2411 → 10000 Len=400
6	0.000118	172.30.42.16	192.168.1.1	UDP	442	2412 → 10000 Len=400
7	0.000130	172.30.42.16	192.168.1.1	UDP	442	2413 → 10000 Len=400
8	0.000142	172.30.42.16	192.168.1.1	UDP	442	2414 → 10000 Len=400
9	0.000154	172.30.42.16	192.168.1.1	UDP	442	2415 → 10000 Len=400
10	0.000167	172.30.42.16	192.168.1.1	UDP	442	2416 → 10000 Len=400

```
▶ Frame 1: 442 bytes on wire (3536 bits), 442 bytes captured (3536 bits) on interface 0 (outbound)
▶ Ethernet II, Src: Apple_b7:2c:89 (f4:5c:89:b7:2c:89), Dst: ArrisGro_f4:f2:34 (44:e1:37:f4:f2:34)
▶ Internet Protocol Version 4, Src: 172.30.42.16 (172.30.42.16), Dst: 192.168.1.1 (192.168.1.1)
▶ User Datagram Protocol, Src Port: 2407 (2407), Dst Port: 10000 (10000)
▼ Data (400 bytes)
    Data: 585858585858585858585858585858585858585858585858...
    [Length: 400]
```

ICMP Attacks

- Ping of Death
  - ☞ Send a ping packet that exceeds maximum size of (64kb) using IP fragmentation
  - ☞ Reassembled packet caused several operating systems to crash due to a buffer overflow
- Smurf IP Attack
  - ☞ Ping a broadcast address using a spoofed source address



PORT KNOCKING

- ✺ Broadly port knocking is the act of attempting to make connections to blocked ports in a certain order in an attempt to open a port
- ✺ Port knocking is fairly secure against brute force attacks since there are 65536k combinations, where k is the number of ports knocked
- ✺ Port knocking however if very susceptible to replay attacks. Someone can theoretically record port knocking attempts and repeat those to get the same open port again
- ✺ One good way of protecting against replay attacks would be a time dependent knock sequence.

IP SPOOFING ATTACKS

- ☞ IP Spoofing is an attempt by an intruder to send packets from one IP address that appear to originate at another
- ☞ If the server thinks it is receiving messages from the real source after authenticating a session, it could inadvertently behave maliciously
- ☞ There are two basic forms of IP Spoofing
    - ☞ Blind Spoofing
        - ☞ Attack from any source
    - ✺ Non-Blind Spoofing
        - ☞ Attack from the same subnet

IP SPOOFING ATTACKS

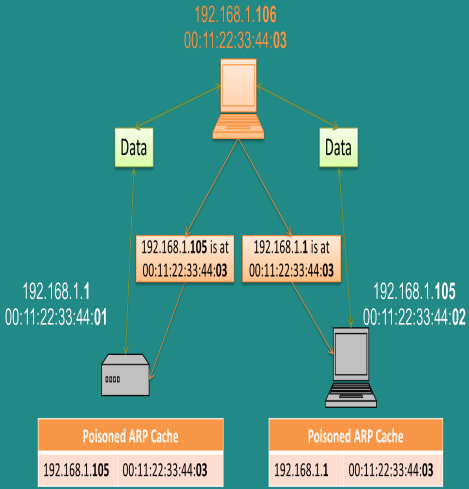
- Blind Spoofing
  - The TCP/IP protocol requires that “acknowledgement” numbers be sent across sessions
  - Need to have the right sequence of acknowledgment numbers to spoof an IP identity
- Non-Blind Spoofing
  - IP Spoofing without inherently knowing the acknowledgment sequence pattern
  - Use a packet sniffer to analyse the sequence pattern
    - decodes and analyses the packets sent across the network
    - Determine the acknowledgment sequence pattern from the packets
    - Send messages to server with actual client’s IP address and with validly sequenced acknowledgment number

MAC SPOOFING

- **MAC filtering**:the network administrator can create a block or allow list of MAC addresses to certain network
- For example on your router at home, you can specify which MAC address (physical address)is allowed to connect and which will be blocked.
- Although this in theory can be useful and it protects your network, in reality it is not very effective.
- If an attacker sniff the traffic of your network, they will be able to identify which device is connected to the network, and hence the relevant MAC address of this device.
- Then the attacker simply spoof their MAC address to the address identified before and hence override any MAC filtering.

ARP Poisoning

- Essentially the attacker needs a way to ”hold down” the client from sending into the session new data that would shift sequence numbers forward.
- To do this, the attacker could just send the data to inject and hope it is received before the real client can send new data. or Dos the Client



192.168.1.**106**
00:11:22:33:44:**03**

Data

Data

192.168.1.**105** is at
00:11:22:33:44:**03**

192.168.1.**1** is at
00:11:22:33:44:**03**

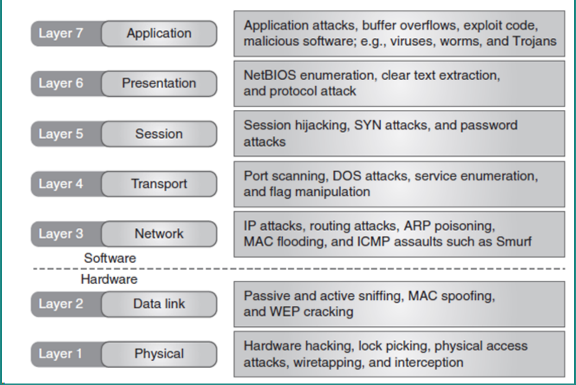
192.168.1.**1**
00:11:22:33:44:**01**

192.168.1.**105**
00:11:22:33:44:**02**

Poisoned ARP Cache	
192.168.1.**105**	00:11:22:33:44:**03**

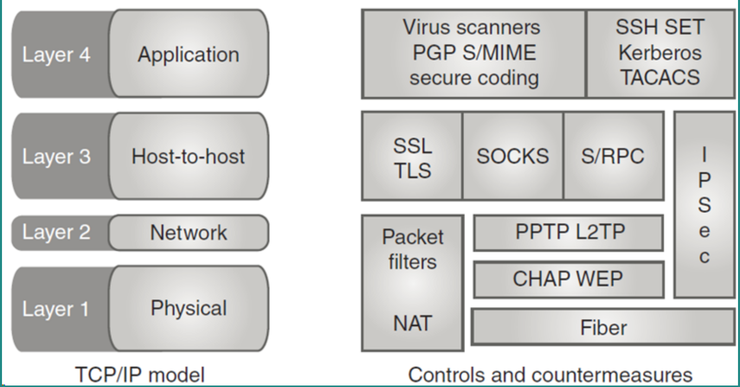
Poisoned ARP Cache	
192.168.1.**1**	00:11:22:33:44:**03**

MAPPING THE OSI MODEL TO CYBER THREATS



Layer 7	Application	Application attacks, buffer overflows, exploit code, malicious software; e.g., viruses, worms, and Trojans
Layer 6	Presentation	NetBIOS enumeration, clear text extraction, and protocol attack
Layer 5	Session	Session hijacking, SYN attacks, and password attacks
Layer 4	Transport	Port scanning, DOS attacks, service enumeration, and flag manipulation
Layer 3	Network	IP attacks, routing attacks, ARP poisoning, MAC flooding, and ICMP assaults such as Smurf
Software		
Hardware		
Layer 2	Data link	Passive and active sniffing, MAC spoofing, and WEP cracking
Layer 1	Physical	Hardware hacking, lock picking, physical access attacks, wiretapping, and interception

Mapping the TCP/IP Model to Security Controls



Layer 4 Application

Layer 3 Host-to-host

Layer 2 Network

Layer 1 Physical

TCP/IP model

Virus scanners
PGP S/MIME
secure coding

SSH SET
Kerberos
TACACS

SSL
TLS

SOCKS

S/RPC

IPSec

Packet
filters

NAT

PPTP L2TP

CHAP WEP

Fiber

Controls and countermeasures

References

- The lecture notes and contents were compiled from my own notes and from various sources.
- Figures and tables are from the recommended books
- **Recommended Readings note:** Focus on what was covered in the class.
  - Chapters 7, 22- Computer Security: Principles and Practice, , William Stallings and Lawrie Brown
  - Chapter 19, CyBOK, The Cyber Security Body of Knowledge