eJPT Certification

Section: Networking

08/7/2020

Learning Objectives:

* Modern network protocols
* Ways computers talk to each other
* Sniffing and capturing the network traffic

**Protocols**:

* Are used in every computer network communication
* In a computer network, machines talk to each other by means of protocols
* The exchange of information between networked computers are called packets.
  + Packets are streams of bits running as electric signals on physical media used for data transmission.
    - This media can be a wire in a LAN (local area network) or the air in a WIFI network.
  + Every packet has the following structure: Header and Payload
    - Header- protocol-specific structure: This ensures that the receiving host can correctly interpret the payload and handle the overall communication
      * The header contains valuable information such as the IP version(version 4 or 6), source address, and destination address.
      * The header allows the nodes involved in the communication to understand and use IP packets.
    - Payload- the actual information. Could be many different types of information such as emails, media, files, etc.

**Protocol Layers:**

* There are many protocols all of which have a specific purpose, such as:
  + Exchanging emails, files or performing VoIP calls.
  + Establishing a communication between a server and a client.
  + Identifying computers on a network.
  + Transmitting data.
* Features that a protocol provides:
  + Make an application (such as an email client, FTP, browser) work.
  + Transport data between processes (the server and the client programs).
  + Identify hosts.
  + Use the physical media to send packets.
* If we rewrite the previous list, it would look like the following:
  + Application layer.
  + Transport layer.
  + Network layer.
  + Physical layer.
* Each of these layers works on top of another, and every layer has its own protocol.
  + Furthermore, each layer serves the one above it (physical supports network, network supports transport, and transport supports application.)

**ISO/OSI:**

* **ISO**- International Organization for Standardization, which in 1984 published a theoretical model for network systems communications: **The Open System interconnection (OSI) model**.
  + **ISO/OSI** was never implemented, but it’s widely used in literature or when talking about IT networks.
  + **Microsoft windows operating systems** use a network architecture that is based on the seven-layer network model developed by ISO.
* OSI has seven layers (every protocol has its own header and payload to help communicate when the next layer)
  + Application
  + Presentation
  + Session
  + Transport
  + Network
  + Data Link
  + Physical
* **Encapsulation** 
  + How do protocols work together?
    - The entire upper protocol packet (header plus payload) is the payload of the lower one; this is **encapsulation.**
  + **IP Protocol suite (TCP/IP)**
    - Protocol stack used on the internet.
    - TCP/IP has four layers:
      * Application
        + App layer gives its packet to the transport layer, which adds its own header.
      * Transport
        + The app packet is now the transport protocol’s payload.
      * Network
        + The transport packet is now handed off to the network layer which adds its own header.
      * Data Link
        + The network packet is now handed off to data link layer where is adds its own header.
    - During encapsulation every protocol adds its own header to the packet, treat it as a payload.
    - This happens to every packet sent by a host.

A screenshot of a cell phone

Description automatically generated

**Internet Protocol (IP):**

* Why is this important?
  + Understand network attacks
  + Using network attack tools at their maximum
  + Studying other networking protocols
* The **Internet Protocol (IP)** is the protocol that runs on the internet layer of the IP suite, also known as TCP/IP.
* IP is tasked w/ delivering the datagrams (IP packets) to the hosts involved in a communication, and it uses IP addresses to ID a host.
* Any host on a computer network, be it a private network or the internet, ID by a **unique IP address.**
* Most networks run IP version (IPv4).
  + IPv4 consists of four bytes, or octets; a byte consists of 8 bits.
    - Ex: 73.5.12,132 (8 bits)
    - Each integer or group of integers separated by “.” Is an octet.
    - With 8 bits you can represent up to 2^8 different values from 0 255.
    - Some addresses are reserved for special purposes.
      * Ex:

A picture containing table

Description automatically generated

**IP/Mask:**

* To correctly ID a host you must know it’s network you will need an IP address and a netmask or subnet mask.
* With an IP/netmask pair, you can accurately ID the network and host parts of the IP address.
  + **A close up of a logo

    Description automatically generated**Example:
* To find the network part you have to perform a bitwise AND operation between the netmask and the IP address.
* Example: 192.168.33.12/255.255.224.0
  + Convert the IP address and subnet mask to their binary form.
    - IP & Mask = Network
    - Which will give you the network prefix in decimal notation: 192.168.32.0
  + 192.168.32.0 is the network prefix. You can now ID the network by using the following notation: 192.168.32.0/255.225.224.0
  + Or, 192.168.32.0/19, because the netmask is made by 19 consecutive “1” bits
    - Known as Classless Inter-Domain Routing (CIDR) notation.
* We can find the host part of the IP address by performing a bitwise AND with the inverse of the netmask.
* IP & Mask = Host
* Which gives you the Host part in decimal notation: 0.0.1.12
* The inverse of the netmask lets us know how many hosts a network can contain.
  + In this example, we have 13 bits to represent the hosts; this means that the network can contain 2^13 = 8192 different addresses.
* There are two special addresses:
  + One with the host part made by all zeros.
  + Another with the host part made by all ones.

**IPv6:**

* **IPv6** address are 128 bits. 2^128 = 2^32 \* 2^96 possible addresses. 2^96 = 79 octillion addresses.
* **IPv6** address consists of 16-bit hexadecimal numbers separated by a colon (:). Hexadecimal numbers are case insensitive. Zeroes can be skipped.
* Examples of IPv6 representation:
  + Regular form: 1080:0:FF:0:8:800:200c:417A
  + Compressed form: FF01:0:0:0:0:0:0:43, which becomes FF::43 as a result of skipping zeros.
  + IPv4- compatible: 0:0:0:0:0:013.1.68.3 or ::13.1.68.3 after skipping zeros.
* IPv6 has reserved addresses which cannot be used.
  + Example: ::1/128 is a loopback address.
  + ::FFFF:0:0/96 are IPv4 mapped addresses.
* IPv6 address can be split in half (64 bits each). Network part and device part.
* The first 64 bits ends with a dedicated 16-bits space (one hex word) that can be used only for specifying a subnet.

**A screenshot of a cell phone

Description automatically generated**

* IPv6 Scope
  + IPv6 has three types of addresses:
    - Global Unicast Address- These addresses are global ones and reside in global internet.
    - Unique Local – Scope Internal Network or VPN- Internally routable but not routed on Internet.
    - Link Local- Scope network link-not routed internally or externally.
  + Addresses can be translated into binary
* IPv6 Subnets
  + IPv6 addresses has a dedicated subnetting portion.
  + Network Address Range- the first 48 bits are for internet global address.
  + Subnetting Range- the 16 bits from the 49th to the 64th are for defining subnets.
  + Device (Interface) Range- the last 64 bits are for device interface ID’s.
* IPv6 Subnetting
  + There are prefixes instead of subnets blocks.
    - Example: 2001:1111:1234:1234::/64
      * The number after the slash (64) is the number of bits that is used for a prefix. Everything behind it can be used for hosts of the subnet.
    - Each 4-digit hex word is 16 bits so the IPv6 address can be divided into groups of four.
    - Example:

**A screenshot of a cell phone

Description automatically generated**

**Routing:**

* Routers are devices connected to different networks at the same time, thus providing a valid path for packets to follow. They are able to forward IP datagrams from one network to another.
  + Forwarding policy is based on **routing protocols**.
* The router inspects the destination address of every incoming packet and then forwards it through one of its interfaces.
* **Routing Table:**
  + The router performs a look up in the routing table, where it finds an IP-to-interface binding.
  + Default address- this entry is used when the router receives a packet whose destination is an *unknown network.*
* Routing Metrics
  + Ensures that if two paths have the same number of hops, the fastest route is selected. The metric is selected according to the channel’s estimated bandwidth and congestion.

**Link Layer Devices and Protocols:**

* Packet forwarding also happens in the lowest layer of the TCP/IP stack: the Link Layer.
  + Link layer devices and protocols only deal with the next hop.
* Hubs and switches are network devices that forward frames (layer 2 packets) on a local network.
  + They work with the link layer network addresses: MAC addresses.
* MAC addresses:
  + Ip addresses are the layer 3 (Network layer) addressing scheme used to identify a host in a network, while MAC addresses uniquely identify a network card (layer 2).
  + MAC (Media Access Control) address is also known as a physical address.
    - 48 bits (6 bytes) long and expressed in hexadecimal form.
  + **A picture containing drawing

    Description automatically generated**Commands too look up MAC address:
    - Ipconfig /all (windows)
    - Ifconfig (MAC)
    - Ip addr Linux
  + Every host on a network has both a MAX and IP address.

**------------------------------------------------- Stopped Taking Notes -----------------------------------------------**

**Notes:**

Review converting the subnet masks to binary equivalent.

**Switches:**

* Steps for packet forwarding
  + The switch reads the destination MAC address of the frame.
  + It performs a look-up in the CAM table.
  + It forwards the packet to the corresponding interface.
  + If there is no entry with that MAC address, the switch will forward the frame to all its interfaces.

ARP (Address Resolution Protocol)

* With ARP a host can build the correct IP address- MAC addresses binding.
* Fundamental protocol for any modern network.
* When a host creates an ARP request it sends a packet containing the destination ID to the switches because it will send the packet to every host.
  + Once the correct destination receives the packet, it will send back a response with it’s MAC address.
* Checking the ARP Cache
  + arp -a on Windows
  + arp on \*nix OS (Mac)
  + ip neighbour on Linux

TCP and UDP:

* When designing a transport layer protocol you must consider limitations.
* Example, **TCP (Transmission Control Protocol)**:
  + Guarantees packet delivery. Because of that, an application that needs a guaranteed delivery will use TCP as the transport protocol.
  + Must be connection oriented. Must establish a connection before transferring data.
* Most used transport protocol on the internet is TCP. Most apps use it and the IP protocol suite often called TCP/IP.
  + Email clients, web browsers and FTP clients are some common apps using TCP
* **UDP (User Datagram Protocol)**
  + Does not guarantee packet delivery.
  + Connectionless.
  + Faster than TCP and provides better throughput (number of packets per second)
    - For example, a glitch in a streamed video or song. (think Spotify and Netflix)

A screenshot of a cell phone

Description automatically generated

**Three Way Handshake:**

* Establishes a connection between two hosts running TCP: Once a three-way handshake is established then the data transmission may begin.
  + The header fields involved in a handshake are:
    - Sequence number
    - Acknowledgement numbers
    - SYN and ACK flags.

**Firewalls:**

* Firewalls can work on different layers of the OSI model
  + Provide different features and protections.
* Firewall features:
  + Packeting sniffing- rules can be setup to filter packets according to specific criteria such as: source IP address, destination IP address, Protocol, Source port, Destination port.
* When firewalls are in place you may notice the following behavior:
  + TCP SYN are sent, but there is no TCP SYN/ACK replies.
  + TCP SYN packets are sent but a TCP RST/ACT reply is received.
  + Packet filters inspect the header of every packet and how to treat it, common action are:
    - Allow- packets are allowed to pass.
    - Drop- drop the packet without any diagnostic message to the packet source host.
    - Deny- do not let the pass, but notify the source host.
  + Packet filtering is not enough to stop layer 7 attacks because any kind of application layer traffic will pass through the firewall.

**Intrusion Detection Systems (IDS):**

* Inspect application payload trying to detect any potential attack.
* A well-configured IDS can detect pretty much every kind of network threat.
* Support firewalls by providing an extra layer of security from mainstream and well-known attack vectors.
* Two main category types:
  + Network Intrusion Detection Systems (NIDS)
    - Inspect network traffic
    - Placed on routers or in networks with high intrusion risk.
  + Host Intrusion Detection Systems (HIDS)
    - Monitor application logs, file-system changes and changes to the operating system configuration.

**Intrusion Prevention Systems (IPS):**

* Can drop malicious request when the threat has a risk classification above pre-defined threshold.

**Network Address Translation (NAT) and IP Masquerading:**

* Two techniques used to provide access to a network from another network.
* The NAT will provide a default gateway, which means it will route internet traffic through it.
  + The NAT device will rewrite the source IP address of every packet setting, thus masquerading the original client’s IP address.
  + A machine on the internet will never know the original client’s IP address.

**DNS :**

* **DNS structure**

**A screenshot of a cell phone screen with text

Description automatically generated**