Cyber security concepts

Sunday, September 12, 2021 3:43 PM

Security	Explanation	Examples
Endpoint	End to end security and usually for user and host dynamics	VPNs, Firewalls, Packet encryption
Data	Broad concept of protections to access the data.	Username and password, encrypting data transfers
Identity management	Limiting access to certain information. Need to know basis. Reduces data compromise	N/A
Database and infrastructure	Physical barriers in place. Hardware security and access	Security system
Cloud	Protecting information stored in cloud	N/A
Mobile	Safely connecting to mobile networks	Password and account protection. VPNs
Disaster recovery	Continuity plans	Backup files/recovery, power generators
End User	Best practices	Don’t share passwords
Data loss prevention	Three separate sets of data stored in two different media types with one being offsite	Backups
Intrusion detection system	Trace where the attack came from and alert	Tools

Threats	Explanation
Hacking	Gain unauthorized access to a system. Many different types of attacks can be used to gain access
Trojans	File or executable that seems legitimate but is a

	virus that affects the system
Phishing	Disguised themselves and try to get password information from users
Spear phishing	Is a specific phishing attack against a specific person
Whale phishing	Is Phishing attacks again company execs
Malware	Wide variety of cyber threats. A lot of the times it's a program that actually has some ill intentions and compromises data
Social engineering	A type of phishing in order to gain login information
Cross site scripting	Common attack that access a persons sensitive information by visiting a website
Ransomware	Lock up company network of server completely and will not release until the ransom is paid
DNS spoofing	Cache poisoning. A corrupt data code is introduced to the resolver of the DNS system. When the malicious code reaches the server, the server ends up returning the wrong information and the hacker can gain access to sensitive data

Attack	explanation
Drive by attack	Hacker looks for insecure websites that could be vulnerable. Cross site scripting
Sql injection	Query from a database that command is inserted into the parameter of the search
Secure socket layer attack	Hacker seeks to exploit the gap between the user command and where the website receives the commands
eaves dropping attack	Intercept data that is in transit
Password	Gaining someone's password
Birthday attack	Replace legitimate message with a fake one using a hash function
MITM	Sit in a location where the data is readable
DDOS	Flood systems with calls and cause denial to log in.

NVD – National Vulnerability Database

The National Vulnerability Database is a website that lists all publicly categorized vulnerabilities. In cybersecurity, vulnerabilities are classified under "**C**ommon **V**ulnerabilities and **E**xposures" (Or CVE for short).

Exploit-DB

Exploit-DB is a resource that we, as hackers, will find much more helpful during an assessment. Exploit-DB retains exploits for software and applications stored under the name, author and version of the software or application.

Rapid7

Rapid7 is a vulnerability research database. The only difference being that this database also acts as an exploit database

GitHub is extremely useful in finding rare or fresh exploits because anyone can create an account and upload – there is no formal verification process like there is with alternative exploit databases

Searchsploit

Searchsploit is a tool that is available on popular pentesting distributions such as Kali Linux. It is also available on the TryHackMe AttackBox. This tool is an offline copy of Exploit-DB, containing copies of exploits on your system.

CIA Triad

Sunday, September 12, 2021 5:13 PM

Confidentiality	Ensuring that the data is accessible to authorized individuals
Integrity	Wholeness and completeness of the information without any alteration except for privileged individuals
Availability	Ability to use the resource or information when it is needed

Disclosure	Sniffing
Alteration	Man in the Middle
Destruction	Vulnerabilities (like DOS attacks)

Cross site scripting

Sunday, September 12, 2021 4:14 PM

https://www.imperva.com/learn/application-security/cross-site-scripting-xss-attacks/

Two major types: stored and reflected
Stored is the more damaging of the two
Malicious script is entered into a web application

Error checking

Tuesday, October 19, 2021 5:45 PM

Parity bit:
1 or a 0 depending on the number of 1s in the message is even or odd

Checksum :
Checksum is the widely used method for the detection of error in data. This method is more reliable than other methods of detection of errors. This approach uses **Checksum Generator** on Sender side and **Checksum Checker** on Receiver side.

CRC :
CRC or Cyclic Redundancy Check is the error detection method to detect the errors and this method is used by upper layer protocols. It contains **Polynomial Generator** on both sender and receiver side. The polynomial generator is of the type x^3+x^2+x+1.

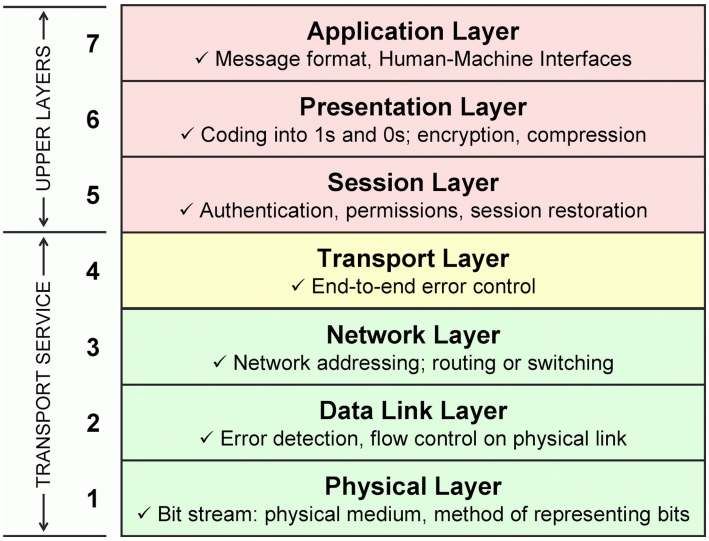
From <https://www.geeksforgeeks.org/difference-between-checksum-and-crc/>

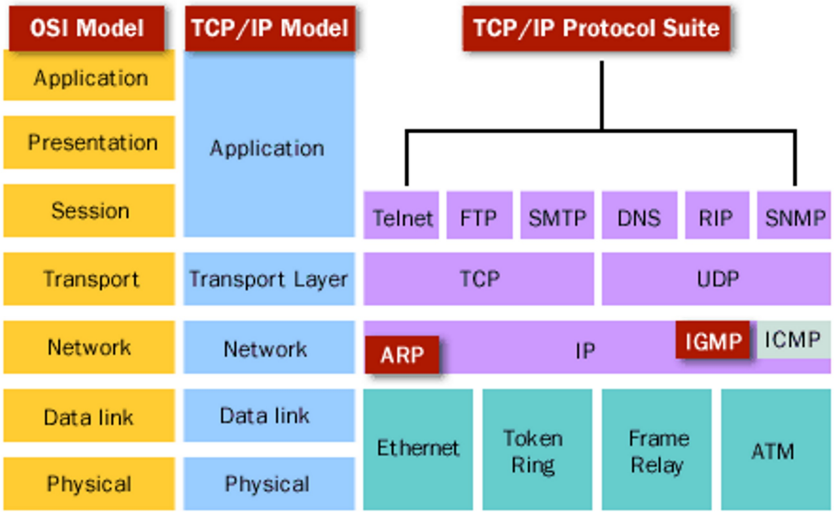
	Checksum	**CRC**
1.	It is not a thorough concept for detection nd reporting of errors.	CRC is a thorough concept for detection and reporting of errors.
2.	It is capable of detecting single bit change in the data.	It is capable of detecting double digits errors.
3.	This method comes after CRC method.	It is the oldest method.
4.	Errors can be easily computed.	It follows a complex computation method.
5.	It can compute less number of errors than CRC.	Due to complex computation, it can detect more errors.
6.	It is based on addition approach.	It is based on hash approach.
7.	It is widely used in data validation during implementation of software.	It is widely used in analog transmission for data validation.

From <https://www.geeksforgeeks.org/difference-between-checksum-and-crc/>

Network Protocols

Tuesday, October 19, 2021 5:58 PM





Telnet

- Port 23
- Old ssh but not secure (not encrypted)
- Can view password in wireshark

Http

- Port 80
- Request html pages and has a handshake mechanism
- Not encrypted
- Need to input http related commands in order to get a response from the http server
  Three popular choices for HTTP servers are:
  - Apache
  - Internet Information Services (IIS)

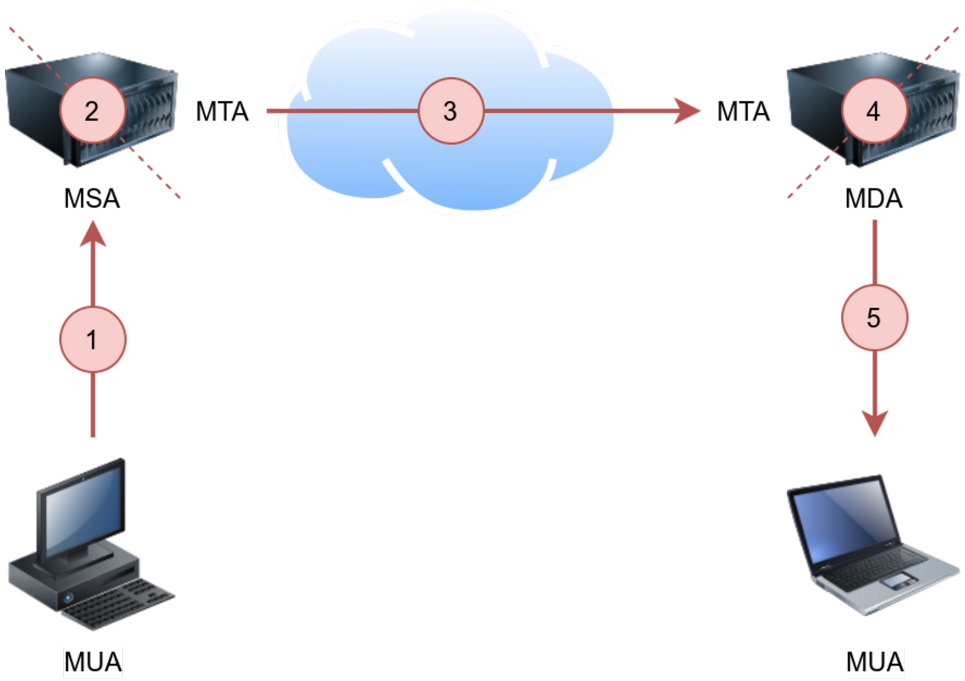
    - nginx
- Nginx and apache are open source
- Web browsers are applications that exchange http easily back and forth
- Format = GET /<filename.html> HTTP/1.1

FTP

- Initialize/listen on port 21
- Login required
- 2 modes of FTP
    - Active: In the active mode, the data is sent over a separate channel originating from the FTP server's port 20.
    - Passive: In the passive mode, the data is sent over a separate channel originating from an FTP client's port above port number 1023.
- File transfer mode
    - Type A = ascii
    - Type I = binary

SMTP, POP3, and IMAP

- Port 25



- Mail User agent (MUA)
    - Client
- Mail Transfer Agent (MTA)
    - Sends over internet. MTA of sender connects to MTA of recipient
- Mail Submission Agent (MSA)
    - Receives message from client and sends it to the MTA
- Mail Delivery Agent (MDA)
    - Recipient collects using their email client
- SMTP
    - Port 25
    - Sending from the MUA to the MSA
- POP3

    - Port 110
    - Downloading mail from MDA
- IMAP
    - Port 143
    - More sophisticated than POP3 since it can sync email accounts on diff devices
    - Not secure since the username and password is sent in clear text

Protocol	TCP Port	Application(s)	Data Security
FTP	21	File Transfer	Cleartext
HTTP	80	Worldwide Web	Cleartext
IMAP	143	Email (MDA)	Cleartext
POP3	110	Email (MDA)	Cleartext
SMTP	25	Email (MTA)	Cleartext
Telnet	23	Remote Access	Cleartext

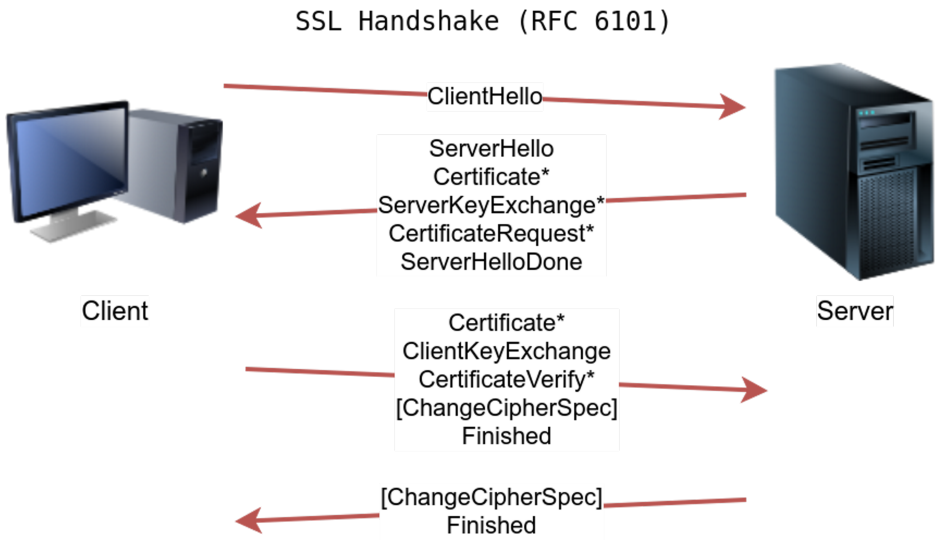
Tcpdump is for quick short captures. Efficient and fast.
Wireshark is more heavy duty. More powerful analysis

TLS

- Use this layer to protect again MITM

Protocol	Default Port	Secured Protocol	Default Port with TLS
HTTP	80	HTTPS	443
FTP	21	FTPS	990
SMTP	25	SMTPS	465
POP3	110	POP3S	995
IMAP	143	IMAPS	993

- HTTPS requires more steps than just regular HTTP
  Establish a TCP connection
  Establish SSL/TLS connection
  Send HTTP requests to the webserver such as GET and POST requests

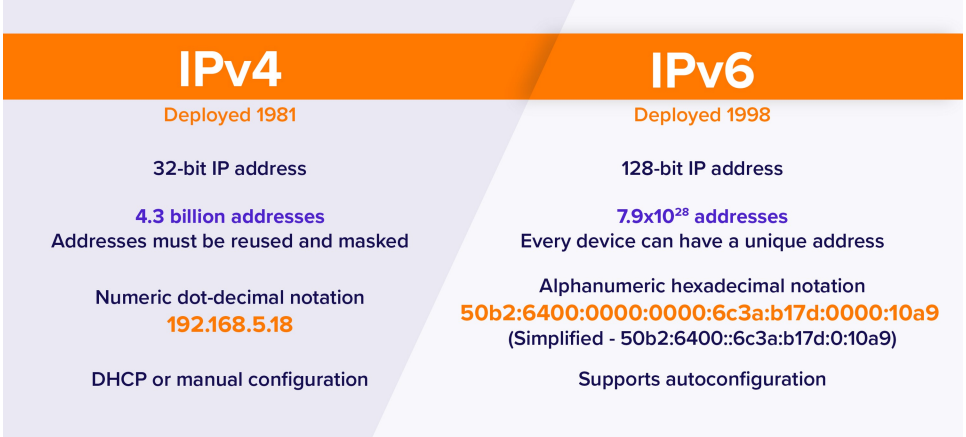


-

The client sends a ClientHello to the server to indicate its capabilities, such as supported algorithms.
The server responds with a ServerHello, indicating the selected connection parameters. The server provides its certificate if server authentication is required. The certificate is a digital file to identify itself; it is usually digitally signed by a third party. Moreover, it might send additional information necessary to generate the master key, in its ServerKeyExchange message, before sending the ServerHelloDone message to indicate that it is done with the negotiation.
The client responds with a ClientKeyExchange, which contains additional information required to generate the master key. Furthermore, it switches to use encryption and informs the server using the ChangeCipherSpec message.
The server switches to use encryption as well and informs the client in the ChangeCipherSpec message.

Nmap, IP, and subnet

Sunday, March 6, 2022 7:05 PM



The advent of IPv6 brought more functionality, in addition to more IP addresses. For example, IPv6 supports **multicast addressing**, which allows bandwidth-intensive packet flows (such as multimedia streams) to be sent to multiple destinations simultaneously, reducing network bandwidth. But is IPv6 better than IPv4? Let's find out.

From <https://www.avast.com/c-ipv4-vs-ipv6-addresses?utm_medium=affiliate&utm_source=commissionjunction&utm_campaign=100357191&utm_content=13305660&couponfield=yes&cjevent=b656682b72a2bc3dd27dd176090a68f477ecc472ad26050f1>

- Subnets with /16, which means that the subnet mask can be written as 255.255.0.0. This subnet can have around 65 thousand hosts.
- Subnets with /24, which indicates that the subnet mask can be expressed as 255.255.255.0. This subnet can have around 250 hosts.

From <https://tryhackme.com/room/nmap01>

If you are connected to the same subnet, you would expect your scanner to rely on ARP (Address Resolution Protocol) queries to discover live hosts

From <https://tryhackme.com/room/nmap01>

PING == ICMP

- Ping request = type 8 echo
- Ping reply = type 0
- Ping timestamp = Type 13
- Ping Timestamp reply = Type 14
- Ping address mask request = type 17
- Ping address mask reply = type 18

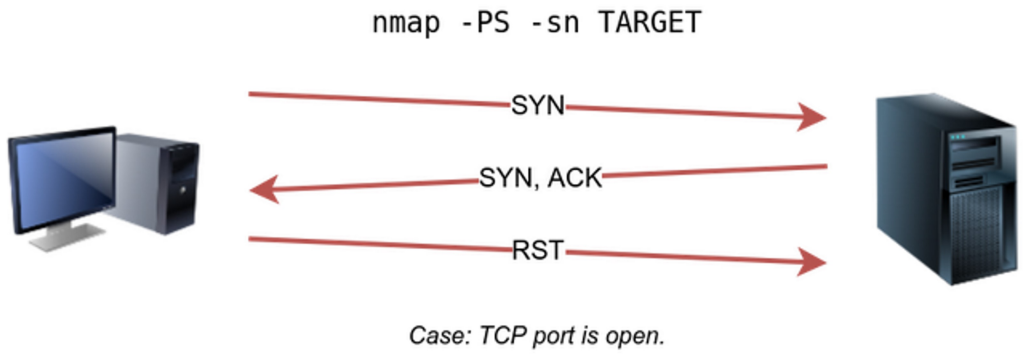
Ping request
https://tryhackme.com/room/nmap01 VIEW SITE BUTTON

- First sends arp request to everyone
- Get arp response from destination comp
- Send a ping request from host
    - Sometimes it can go to router to hop onto a different network
    - In that case, the arp response will come back from router
    - Send ping request to router
    - Router sends arp request on new subnet
    - Destination computer responds and sends arp response to router
    - Router forwards the ping request
- Get ping response from target
    - If there was a router in between, the response will be tunneled to the original requester

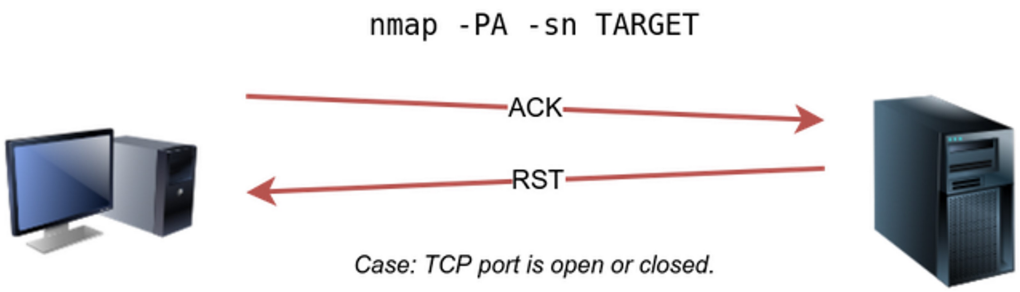
Arp request

- Arp requests only work if youre on the same subnet

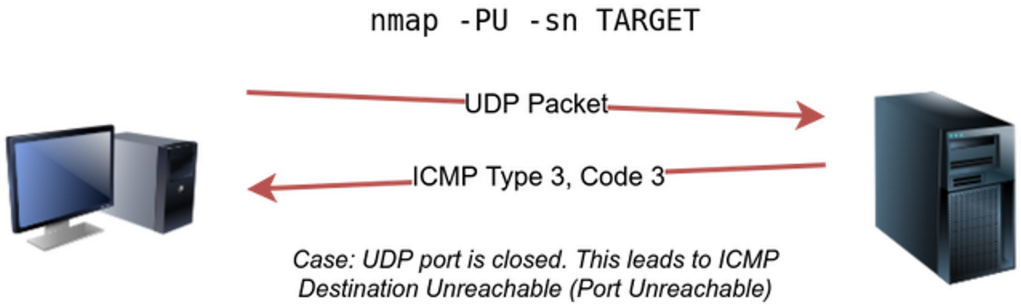
TCP and UDP pings/ test



Case: TCP port is open.



Case: TCP port is open or closed.



Case: UDP port is closed. This leads to ICMP Destination Unreachable (Port Unreachable)

Scan Type	**Example Command**
ARP Scan	sudo nmap -PR -sn MACHINE_IP/24
ICMP Echo Scan	sudo nmap -PE -sn MACHINE_IP/24

ICMP Timestamp Scan	sudo nmap -PP -sn MACHINE_IP/24
ICMP Address Mask Scan	sudo nmap -PM -sn MACHINE_IP/24
TCP SYN Ping Scan	sudo nmap -PS22,80,443 -sn MACHINE_IP/30
TCP ACK Ping Scan	sudo nmap -PA22,80,443 -sn MACHINE_IP/30
UDP Ping Scan	sudo nmap -PU53,161,162 -sn MACHINE_IP/30

From <https://tryhackme.com/room/nmap01>

Option	Purpose
-n	no DNS lookup
-R	reverse-DNS lookup for all hosts
-sn	host discovery only

From <https://tryhackme.com/room/nmap01>

This room covered three types of scans.

Port Scan Type	Example Command
TCP Connect Scan	nmap -sT 10.10.143.240
TCP SYN Scan	sudo nmap -sS 10.10.143.240
UDP Scan	sudo nmap -sU 10.10.143.240

These scan types should get you started discovering running TCP and UDP services on a target host.

Option	Purpose
-p-	all ports
-p1-1023	scan ports 1 to 1023
-F	100 most common ports
-r	scan ports in consecutive order
-T<0-5>	-T0 being the slowest and T5 the fastest
--max-rate 50	rate <= 50 packets/sec
--min-rate 15	rate >= 15 packets/sec
--min-parallelism 100	at least 100 probes in parallel

From <https://tryhackme.com/room/nmap02>

- Null scan
  - No flags
  - If port closed, we get a response
  - No response means port Open|filtered
- Fin scan
  - Fin flag
  - Same as null
  - No response means port open|filtered
- Xmas scan

    - Fin, psh, urg flags
    - Same as null
    - No response means port open|filtered
- Ack scan
    - Ack flag
    - Sees if firewall filters the port
- Window scan
    - Examines tcp windows field
    - If behind a firewall, sees which ports are closed

Spoofing

- Pretending to be a different IP address
- Good if MITM attack on spoofed device
- Can add multiple recipients and just have one of them return back to you

Zombie scan

- Find a zombie machine like a printer on the network and get the IP

Pen Testing steps

Monday, March 7, 2022 12:45 AM

Common Vulnerability Scoring System

1. How easy is it to exploit the vulnerability?
2. Do exploits exist for this?
3. How does this vulnerability interfere with the CIA triad?

Rating	Score
None	0
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Vulnerability Priority Rating (VPR)

This framework is considered to be risk-driven; meaning that vulnerabilities are given a score with a heavy focus on the risk a vulnerability poses to the organization itself, rather than factors such as impact (like with CVSS).

Stage	Description
Information Gathering	This stage involves collecting as much publicly accessible information about a target/organization as possible, for example, OSINT and research. **Note:** This does not involve scanning any systems.
Enumeration/Scanning	This stage involves discovering applications and services running on the systems. For example, finding a web server that may be potentially vulnerable.
Exploitation	This stage involves leveraging vulnerabilities discovered on a system or application. This stage can involve the use of public exploits or exploiting application logic.
Privilege Escalation	Once you have successfully exploited a system or application (known as a foothold), this stage is the attempt to expand your access to a system. You can escalate horizontally and vertically, where horizontally is accessing another account of the same permission group (i.e. another user), whereas vertically is that of another permission group (i.e. an administrator).
Post-exploitation	This stage involves a few sub-stages: **1.** What other hosts can be targeted (pivoting) **2.** What additional information can we gather from the host now that we are a privileged user **3.** Covering your tracks **4.** Reporting

From <https://tryhackme.com/room/pentestingfundamentals>

- **Open-box pen test** - In an open-box test, the hacker will be provided with some information ahead of time regarding the target company's security info.
- **Closed-box pen test** - Also known as a 'single-blind' test, this is one where the hacker is given no background information besides the name of the target company.
- **Covert pen test** - Also known as a 'double-blind' pen test, this is a situation where almost no one in the company is aware that the pen test is happening, including the IT and security professionals who will be responding to the attack. For covert tests, it is especially important for the hacker to have the scope and other details of the test in writing beforehand to avoid any problems with law enforcement.
- **External pen test** - In an external test, the ethical hacker goes up against the company's external-facing technology, such as their website and external network servers. In some cases, the hacker may not even be allowed to enter the company's building. This can mean conducting the attack from a remote location or carrying out the test from a truck or van parked nearby.
- **Internal pen test** - In an internal test, the ethical hacker performs the test from the company's internal network. This kind of test is useful in determining how much damage a disgruntled employee can cause from behind the company's firewall.

From <https://www.cloudflare.com/learning/security/glossary/what-is-penetration-testing/>

1. OSSTMM

The OSSTMM framework, one of the most recognized standards in the industry, provides a scientific methodology for network penetration testing and vulnerability assessment. This framework contains a comprehensive guide for testers to identify security vulnerabilities within a network (and its components) from various potential angles of attack. This methodology relies on the tester's in-depth knowledge and experience, as well as human intelligence to interpret the identified vulnerabilities and their potential impact within the network.

Unlike the majority of security manuals, this framework was also created to support network development teams. A majority of developers and IT teams base their firewalls and networks on this manual and the guidelines it provides. While this manual does not advocate for a particular network protocol or software, it highlights the best practices and the steps that should be taken to ensure the security of your networks.

The OSSTMM methodology (Open Source Security Testing Methodology Manual) allows testers to customize their assessment to fit the specific needs or the technological context of your company. With this set of standards, you will obtain an accurate overview of your network's cybersecurity, as well as reliable solutions adapted to your technological context to help your stakeholders make the right decisions to secure your networks.

2. OWASP

For all matters of application security, the Open Web Application Security Project (OWASP) is the most

recognized standard in the industry. This methodology, powered by a very well-versed community that stays on top of the latest technologies, has helped countless organizations to curb application vulnerabilities.

This framework provides a methodology for web application penetration testing that can not only identify vulnerabilities commonly found within web and mobile applications, but also complicated logic flaws that stem from unsafe development practices. The updated guide provides comprehensive guidelines for each penetration testing method, with over 66 controls to assess in total, allowing testers to identify vulnerabilities within a wide variety of functionalities found in modern applications today.
With the help of this methodology, organizations are better equipped to secure their applications – web and mobile alike – from common mistakes that can have a potentially critical impact on their business.
Organizations looking to develop new web and mobile applications should also consider incorporating these standards during their development phase to avoid introducing common security flaws.
During an application security assessment, you should expect the OWASP standard to be leveraged to ensure that no vulnerabilities have been left behind and that your organization obtains realistic recommendations adapted to the specific features and technologies used in your applications.

From <https://www.vumetric.com/blog/top-penetration-testing-methodologies/>

3. NIST

Unlike other information security manuals, NIST offers more specific guidelines for penetration testers to follow. The National Institute of Standards and Technology (NIST) provides a manual that is best suited to improve the overall Cybersecurity of an organization. The most recent version, 1.1, places more emphasis on the Critical Infrastructure Cybersecurity. Complying with the NIST framework is often a regulatory requirement for various American providers and business partners.

With this framework, NIST set its sight on guaranteeing information security in different industries, including banking, communications, and energy. Large and small firms alike can tailor the standards to meet their specific needs.
In order to meet the standards that NIST has set, companies most perform penetration tests on their applications and networks following a pre-established set of guidelines. This American information tech security standard ensures that companies fulfill their cybersecurity control and assessment obligations, mitigating risks of a cyberattack in every way possible.
Stakeholders from different sectors collaborate to popularize the Cybersecurity Framework and encourage firms to implement it. With exceptional standards and technology, NIST significantly contributes to cybersecurity innovation in a host of American industries.

4. PTES

The PTES Framework (Penetration Testing Methodologies and Standards) highlights the most recommended approach to structure a penetration test. This standard guides testers on various steps of a penetration test including initial communication, gathering information, as well as the threat modeling phases.

Following this penetration testing standard, testers acquaint themselves with the organization and their technological context as much as possible before they focus on exploiting the potentially vulnerable areas, allowing them to identify the most advanced scenarios of attacks that could be attempted. The testers are also provided with guidelines to perform post-exploitation testing if necessary, allowing them to validate that the previously identified vulnerabilities have been successfully fixed. The seven phases provided in this standard guarantee a successful penetration test offering practical recommendations that your management team can rely on to make their decisions.

5. ISSAF

The ISSAF standard (Information System Security Assessment Framework) contains an even more structured and specialized approach to penetration testing than the previous standard. If your organization's unique situation requires an advanced methodology entirely personalized to its context, then this manual should prove useful for the specialists in charge of your penetration test.

These sets of standards enable a tester to meticulously plan and document every step of the penetration testing procedure, from planning and assessment to reporting and destroying artifacts. This standard caters for all steps of the process. Pentesters who use a combination of different tools find ISSAF especially crucial as they can tie each step to a particular tool.
The assessment section, which is more detailed, governs a considerable part of the procedure. For each vulnerable area of your system, ISSAF offers some complementary information, various vectors of attack, as well as possible results when a vulnerability is exploited. In some instances, testers may also find information on tools that real attackers commonly use to target these areas. All this information proves worthwhile to plan and carry out particularly advanced attack scenarios, which guarantees a great return on investment for a company looking to secure their systems from cyberattacks.

From <https://www.vumetric.com/blog/top-penetration-testing-methodologies/>

Attack types

Sunday, April 24, 2022 12:28 PM

Principle	Description
Spoofing	This principle requires you to authenticate requests and users accessing a system. Spoofing involves a malicious party falsely identifying itself as another. Access keys (such as API keys) or signatures via encryption helps remediate this threat.
Tampering	By providing anti-tampering measures to a system or application, you help provide integrity to the data. Data that is accessed must be kept integral and accurate. For example, shops use seals on food products.
Repudiation	This principle dictates the use of services such as logging of activity for a system or application to track.
Information Disclosure	Applications or services that handle information of multiple users need to be appropriately configured to only show information relevant to the owner is shown.
Denial of Service	Applications and services use up system resources, these two things should have measures in place so that abuse of the application/service won't result in bringing the whole system down.
Elevation of Privilege	This is the worst-case scenario for an application or service. It means that a user was able to escalate their authorization to that of a higher level i.e. an administrator. This scenario often leads to further exploitation or information disclosure.

From <https://tryhackme.com/room/principlesofsecurity>

Vulnerability exploits

Vulnerability	Description
Operating System	These types of vulnerabilities are found within Operating Systems (OSs) and often result in privilege escalation.
(Mis) Configuration-based	These types of vulnerability stem from an incorrectly configured application or service. For example, a website exposing customer details.
Weak or Default Credentials	Applications and services that have an element of authentication will come with default credentials when installed. For example, an administrator dashboard may have the username and password of "admin". These are easy to guess by an attacker.
Application Logic	These vulnerabilities are a result of poorly designed applications. For example, poorly implemented authentication mechanisms that may result in an attacker being able to impersonate a user.
Human-Factor	Human-Factor vulnerabilities are vulnerabilities that leverage human behaviour. For example, phishing emails are designed to trick humans into believing they are legitimate.

From <https://tryhackme.com/room/vulnerabilities101>

Vulnerability	Description

Security Misconfigurations	Security misconfigurations involve vulnerabilities that are due to developer oversight. For example, exposing server information in messages between the application and an attacker.
Broken Access Control	This vulnerability occurs when an attacker is able to access parts of an application that they are not supposed to be able to otherwise.
Insecure Deserialization	This is the insecure processing of data that is sent across an application. An attacker may be able to pass malicious code to the application, where it will then be executed.
Injection	An Injection vulnerability exists when an attacker is able to input malicious data into an application. This is due to the failure of not ensuring (known as sanitizing) input is not harmful.

A foothold is an access to the vulnerable machine's console, where we can then begin to exploit other applications or machines on the network.

MITM attacks
Many tools would aid you in carrying out such an attack, such as Ettercap and Bettercap.

Password Attacks
We want an automated way to try the common passwords or the entries from a word list; here comes THC Hydra. Hydra supports many protocols, including FTP, POP3, IMAP, SMTP, SSH, and all methods related to HTTP. The general command-line syntax is:
Hydra remains a very efficient tool that you can launch from the terminal to try the different passwords. We summarize its main options in the following table.

Option	**Explanation**
-l username	Provide the login name
-P WordList.txt	Specify the password list to use
server service	Set the server address and service to attack
-s PORT	Use in case of non-default service port number
-V or -vV	Show the username and password combinations being tried
-d	Display debugging output if the verbose output is not helping

From <https://tryhackme.com/room/protocolsandservers2>

Network pen testing

Sunday, April 24, 2022 2:06 PM

WHOIS protocol

- to get various information about the domain name we were looking up. In particular, we were able to get the DNS servers from the registrar.

NSLOOKUP

- Find the IP address of a domain name using nslookup, which stands for Name Server Look Up.
- Type=MX is the email servers that are used by the domain

DIG

- For more advanced DNS queries and additional functionality, you can use dig

DNSDumpster

- Find subdomains for more information
- https://dnsdumpster.com/

Shodan.io

- helpful to learn various pieces of information about the client's network, without actively connecting to it. Furthermore, on the defensive side, you can use different services from Shodan.io to learn about connected and exposed devices belonging to your organization.

Summary

Purpose	**Commandline Example**
Lookup WHOIS record	whois tryhackme.com
Lookup DNS A records	nslookup -type=A tryhackme.com
Lookup DNS MX records at DNS server	nslookup -type=MX tryhackme.com 1.1.1.1
Lookup DNS TXT records	nslookup -type=TXT tryhackme.com
Lookup DNS A records	dig tryhackme.com A
Lookup DNS MX records at DNS server	dig @1.1.1.1 tryhackme.com MX
Lookup DNS TXT records	dig tryhackme.com TXT

From <https://tryhackme.com/room/passiverecon>

Add-ons for web browser

- **FoxyProxy** lets you quickly change the proxy server you are using to access the target website. This browser extension is convenient when you are using a tool such as Burp Suite or if you need to switch proxy servers regularly. You can get FoxyProxy for Firefox from here.
- **User-Agent Switcher and Manager** gives you the ability to pretend to be accessing the webpage from a different operating system or different web browser. In other words, you can pretend to be browsing a site using an iPhone when in fact, you are accessing it from Mozilla Firefox. You can download User-Agent Switcher and Manager for Firefox here.
- **Wappalyzer** provides insights about the technologies used on the visited websites. Such extension is handy, primarily when you collect all this information while browsing the website like any other user. A screenshot of Wappalyzer is shown below. You can find Wappalyzer for Firefox here.

Traceroute

- The number of hops/routers between your system and the target system depends on the time you are running traceroute. There is no guarantee that your packets will always follow the same route, even if

you are on the same network or you repeat the traceroute command within a short time.

- Some routers return a public IP address. You might examine a few of these routers based on the scope of the intended penetration testing.
- Some routers don't return a reply

Netcat

Netcat supports both TCP and UDP protocols. It can function as a client that connects to a listening port; alternatively, it can act as a server that listens on a port of your choice. Hence, it is a convenient tool that you can use as a simple client or server over TCP or UDP.

Active recon

You can use traceroute to map the path to the target, ping to check if the target system responds to ICMP Echo, and telnet to check which ports are open and reachable by attempting to connect to them. Available scanners do this at much more advanced and sophisticated levels, as we will see in the next four rooms with nmap.

Command	**Example**
ping	ping -c 10 10.10.175.213 on Linux or macOS
ping	ping -n 10 10.10.175.213 on MS Windows
traceroute	traceroute 10.10.175.213 on Linux or macOS
tracert	tracert 10.10.175.213 on MS Windows
telnet	telnet 10.10.175.213 PORT_NUMBER
netcat as client	nc 10.10.175.213 PORT_NUMBER
netcat as server	nc -lvnp PORT_NUMBER

Although these are fundamental tools, they are readily available on most systems. In particular, a web browser is installed on practically every computer and smartphone and can be an essential tool in your arsenal for conducting reconnaissance without raising alarms. If you want to gain more profound knowledge of the Developer Tools, we recommend joining Walking An Application.

Operating System	**Developer Tools Shortcut**
Linux or MS Windows	Ctrl+Shift+I
macOS	Option + Command + I

Metasploit

Thursday, May 19, 2022 9:11 PM

Pro
- Automation and task management (with gui)

Framework
- Open source pen testing (command line)

Metasploit framework
- Msfconsole
  - Command line interface
- Modules
  - Exploits, scanners and payloads
- Tools
  - Uses modules

Concepts
- Exploit
  - Uses a vulnerability
- Vulnerability
  - Design, code, or logic flaw on target system
- Payload
  - Code that runs the exploit on target system

Modules
- Auxilary
  - Supporting module like scanners crawlers and fuzzers
- Evasion
  - Encode the payload but not for antivirus software
- Exploits
  - Organized by target
- NOPs
  - Do nothing cycle(for buffering)
- Payloads
  - Codes that will run on target
  - Singles
    - Self contained that don't need downloads
  - Stagers
    - Setting up connection between attacker and target
  - Stages
    - Downloaded by stager to use larger sized payloads
- Post
  - Useful after exploiting

Msfconsole
- Use
  - In order to use an exploit
- Show options
  - All of the settings for the exploit selected
- Show payloads
  - All of the versions of payloads for the exploit

- Back
    - Exit the use functionality of the payload
- SEARCH (one of the most useful)
    - To find the exploit names or payloads based on cve numbers or target

EXPLOITING

Port scanning
- Metasploit has port scan scripts
- Or run nmap on metasploit
- Fields
    - Concurrency - number of targets in parallel
    - Ports - port range to be scanned
    - Rhosts - list of targets
    - Threads - threads to be use simultaneously

Udp service ID
- Quick scan to identify common udp such as DNS/NetBIOS

SMB Scan
- For server/shard files on windows
- Can run to find windows versions if port is open
- Smb_login to guess password of username

Metasploit DB/workspace
- Database allow isolation of projects called workspaces
- Like python environments
- Db_nmap will store nmap results in the database
- Hosts and service command will list the hosts scanned and the services of those hosts that have been scanned
- Low hanging fruits to look into
    - HTTP: Could potentially host a web application where you can find vulnerabilities like SQL injection or Remote Code Execution (RCE).
    - FTP: Could allow anonymous login and provide access to interesting files.
    - SMB: Could be vulnerable to SMB exploits like MS17-010
    - SSH: Could have default or easy to guess credentials
    - RDP: Could be vulnerable to Bluekeep or allow desktop access if weak credentials were used.

From <https://tryhackme.com/room/metasploitexploitation>

Exploits
- Set payload to choose which payload to use
- Sessions will show which exploited sessions you have open

Python Notes

Tuesday, October 19, 2021 5:33 PM

Basic notes

- python3 print statements always have '()' followed
- inputs are treated as strings 'var = input("enter in a val:")'
- results of ints cut off the decimal values
- raise IOError("file error")
- except Myerror as err:
- var = (letter for letter in 'hello world') returns a generator or iter()
- interpreted means doesnt need to be compiled before running
- array.count(instances) will count how many instances are in the list
- list[beginning:end:step increments]
- append adds to end of list and pop will take from the end
- keys and values of dicts

intermediate notes

- arg is the normal. argv is an number of inputs. list of inputs
- kwargs** is when the inputs have a key value assigned kwargs. dict of inputs
- pdb.set_trace() in the code in order to go through it python-m pdb script.py
- generator is an interator that you can only iterate through once
- generator are best for large sets that you dont want to allocate space. Faster and more space efficient
- Map apply function to all items in a list
- Filter only leaves the true statement given
- Reduce does computation on a list
- short hand ternary is msg = output or "No data returned"
- Functions that modify the functionality of other functions
- () executes the function
- decorator is the function that does code before and after an input function
- global can be defined anywhere as global val (dont usually use)
- namedtuple is like a dict
- mutable references and instantiations are confusing
- __slots__ for class vars to use less ram
- virtualenv is isolated environments
- virtualenv myproject
- source myproject/bin/activate
- zip will make a tuple of all of the elements according to index
- type wil return type and id will return the unique id of object
- comprehension is the forloop to populate collection objects
- try: and then except Exception as e: will catch all of it
- the finally will run whether or not the exception occured
- the else will run if no exceptions were encountered
- multiple dunder methods like __init__ or __getitem__
- swig interfaces c code into python

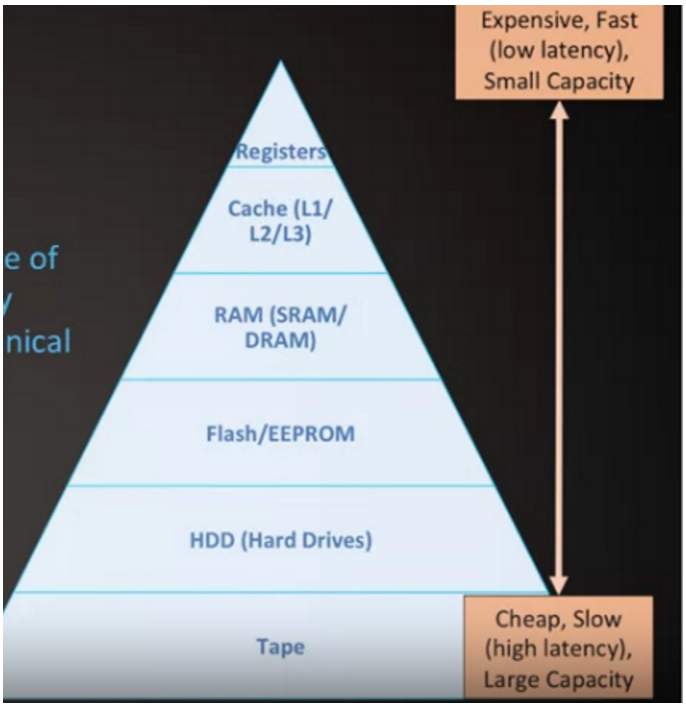
Useful Things to remember

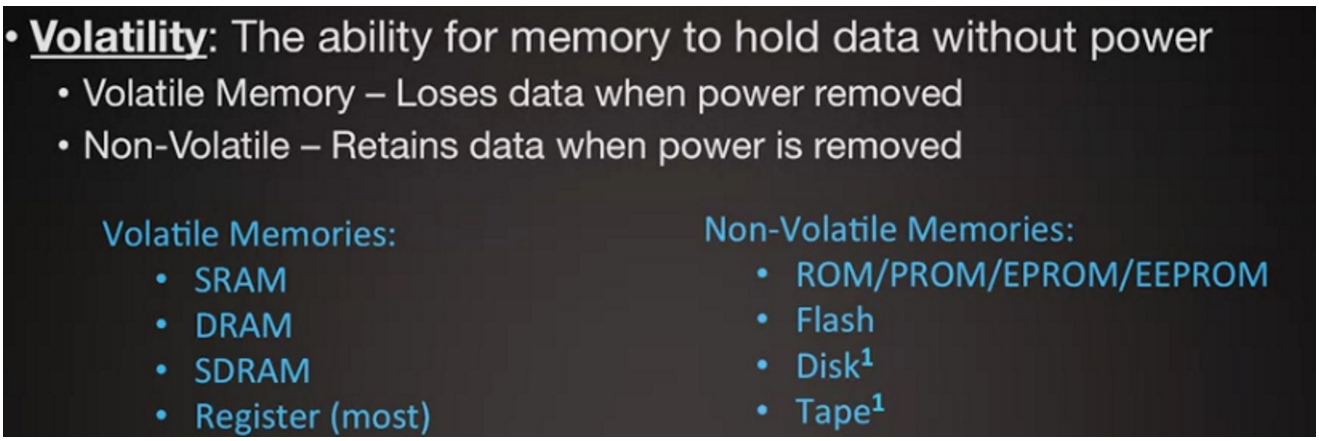
- Enumerate creates a pair (tuple) of the array and the index of array element. (returns as (index, element))
- intervals.sort(key=lambda x: x[0])
    - If you have an array of arrays and need to sort by a certain index
    - The sort method
- Recursive functions can be built/written immediately INSIDE the method that will call the recursive function

```
Def initialmethod(input):
    Def recursiveMethod(inner_input):
        #define the inside
        #call of recursive function
        Return recursiveMethod(modified_inner_input)
    #first call of recursiveMethod
    recursiveMethod(input)
```

Embedded

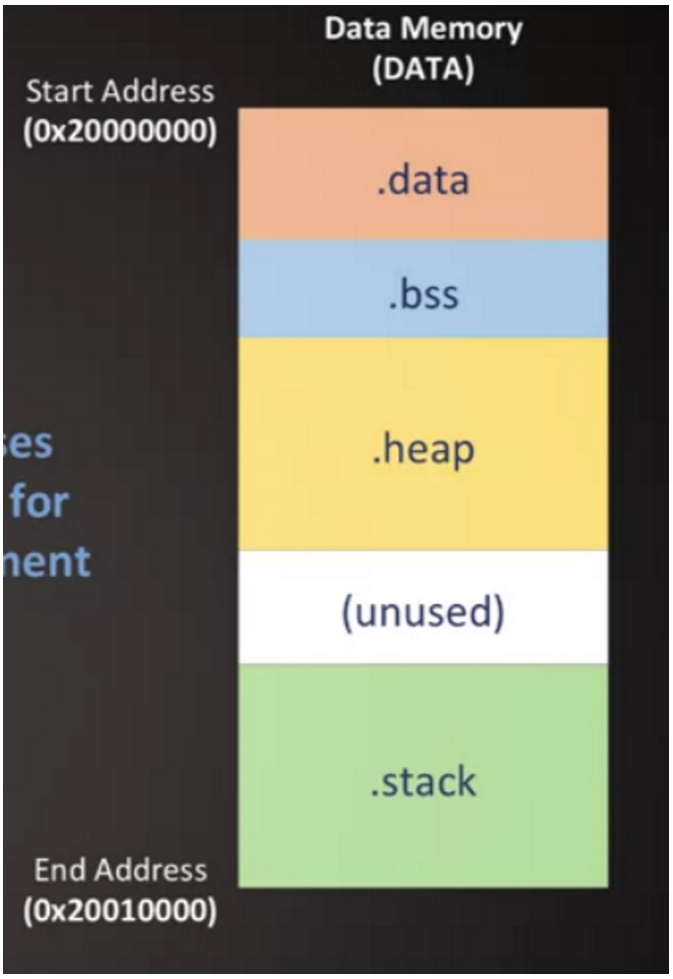
Tuesday, May 3, 2022 10:45 PM



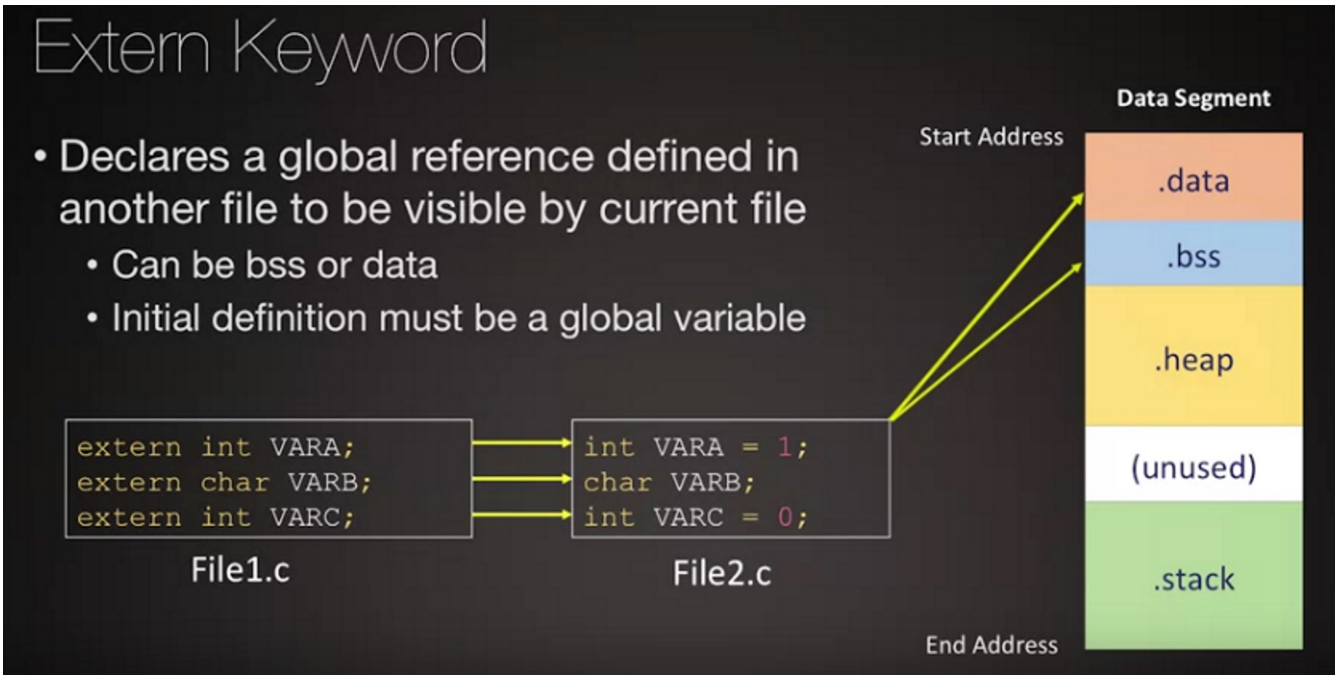


Data memory

- Variables
- Loaded into registers and stored back into memory
- Stack
  - Stores local variables
- Heap
  - Stores dynamic data such as malloc and calloc data
- Bss
  - Zero initialized or Uninitialized global variables
- Data
  - Nonzero initilaized global variables

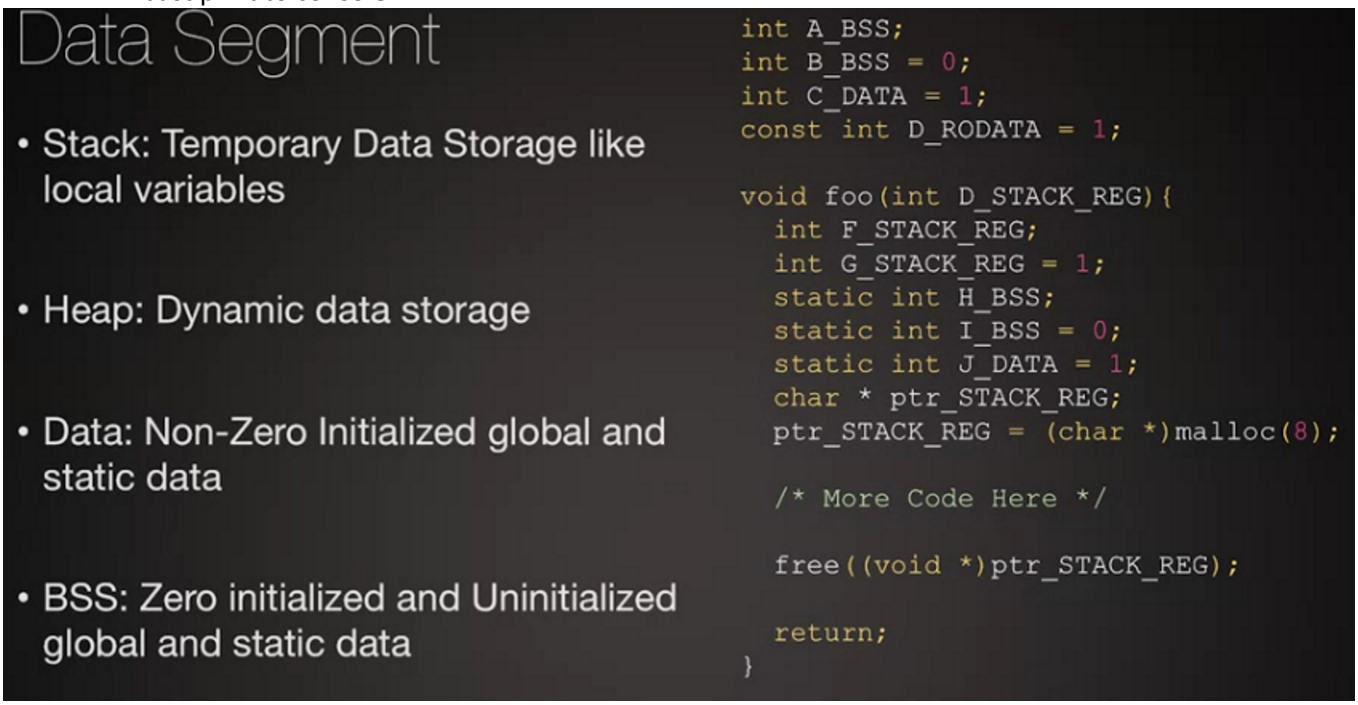


- Text/code
    - Read only so that the program doesn't accidentally write over it or change it



- Scanf
    - Based on the format provided
- Printf

- Just print to console



- There is a mistake on the heap of the mallocs and free of the pointer

Questions

Tuesday, October 19, 2021 7:02 PM

What is DNS and does it use udp or tcp

DNS is PORT 53

DNS uses UDP but can convert to TCP using a zone transfer. Or if the message is too large then it will set it up as a TCP connection to transfer the data.

How do you end a process that's not running correctly

Use the Kill command on the ID number. The kill-9 does an unsafe end to completely kill the command without letting it wrap up and perform its exiting protocol

If you were to perform a security analysis on a web application that uses a database and has an environment accessible by anyone, how would you perform the analysis.

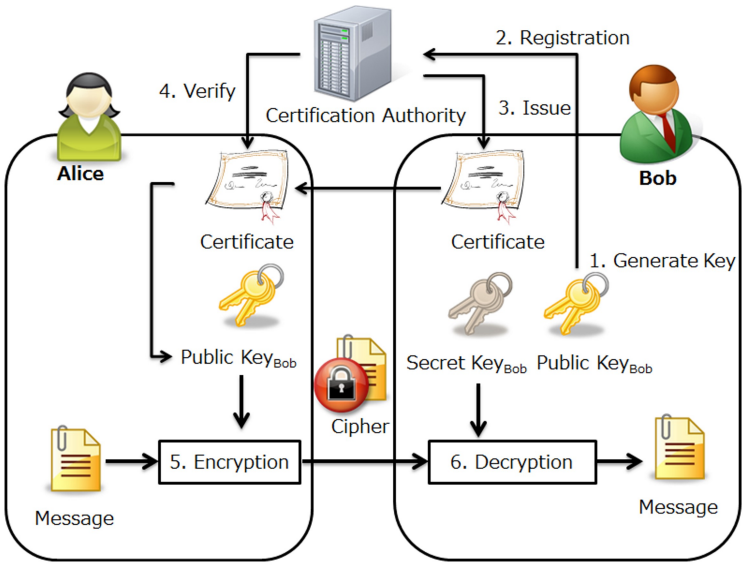
First step is reconnaissance. Finding deeply about the system by footprinting, enumerating, and scanning
See what versions of apache is running on the servers the exact devices and racks that are being used.

And then see system setup If they do any hashing any passwords and if they do encryption on the data that is stored on the servers

Arp tables and what can you make out of them. The first set of mac address has the manufacturer ID and then figure out the device from the rest

What can you tell from the subnet mask.
How many hosts are in the network

The difference between nmap and wireshark. For nmap has an active mode that you can trigger to do you packet captures while wireshark is all passive. Wireshark also doesn't go down to the granularity that nmap can go through to.



Benchmark testing standards.
A **Benchmark in Performance Testing** is a metric or a point of reference against which software products or services can be compared to assess the quality measures. In other words, Benchmark means a set standard that helps to determine the quality of software product or service.

From <https://www.guru99.com/benchmark-testing.html>

What is hadoop used for. And how fast is it
In comparison with traditional computing, yes! Hadoop is fast. Also, Hadoop handles data through clusters, thus, it runs on the principle of the distributed file system, and hence, provides faster processing.
Apache Hadoop is an open-source software utility that allows users to manage big data sets (from gigabytes to petabytes) by enabling a network of computers (or "nodes") to solve vast and intricate data problems. It is a highly scalable, cost-effective solution that stores and processes structured, semi-structured and unstructured data (e.g., Internet clickstream records, web server logs, IoT sensor data, etc.).

Why Hadoop is so popular in the industry?
Why Hadoop is lightning fast as compared to the traditional system?

From <https://data-flair.training/forums/topic/why-hadoop/>

Hadoop was the best solution for storing and processing big data because:
1. It stores huge files as they are (raw) without specifying any schema.
2. High scalability – any number of nodes can be added at once hence enhancing performance dramatically.

3. It’s economic so it suits the purse of anyone starting from a startup to a tech giant. Commodity hardware can be efficiently used with Hadoop.
4. Reliable – As there is no danger of losing data even if nodes in the clusters fail, it’s highly reliable. Recovery and backup of data are automatic.
5. Open source – No headache of licensing. Download and enjoy the power of Hadoop.
The above and many more interesting and useful characteristics of Hadoop combined make it so popular in the industry.
Hadoop is lightning fast because of data locality – move computation to data rather than moving the data, as it is easier and make processing lightning fast. The Same algorithm is available for all the nodes in the cluster to process on chunks of data stored in them. So data processing is not done on one big piece of data, but rather smaller distributed pieces thus enhancing performance by processing them distributedly.

From <https://data-flair.training/forums/topic/why-hadoop/>

From <https://www.ibm.com/cloud/blog/hadoop-vs-spark>

From <https://intellipaat.com/community/37645/is-hadoop-fast-or-slow>

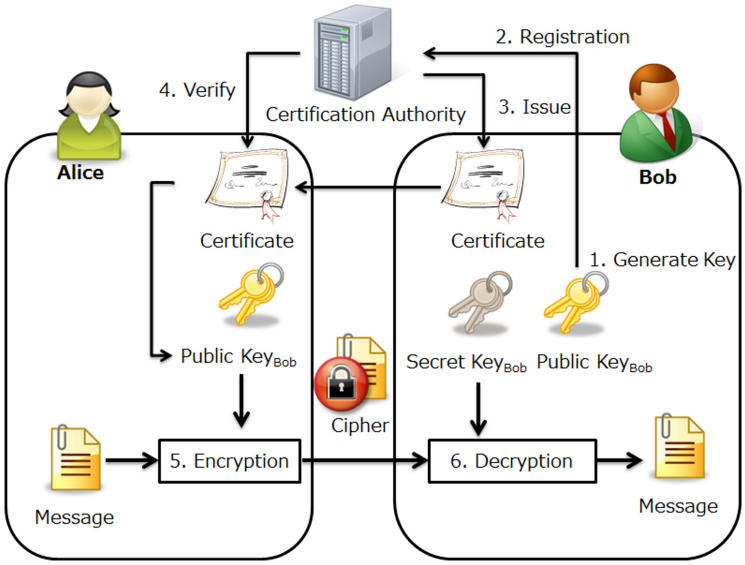
PKI Certificates

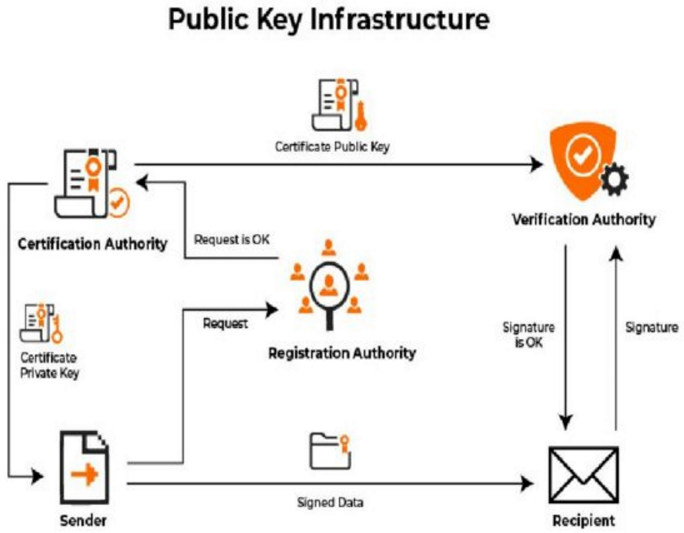
Monday, March 7, 2022 4:30 PM

Pki certificates

PKI is best utilized for situations that require digital security, which is where encryption plays a vital role. PKI performs encryption directly through the keys that it generates. It works by using two different cryptographic keys: a public key and a private key. Whether these keys are public or private, they encrypt and decrypt secure data.

From <https://www.venafi.com/education-center/pki/how-does-pki-work>





1.

2. Digital Certificates

PKI functions because of digital certificates. A digital certificate is like a drivers license—it's a form of electronic identification for websites and organizations. Secure connections between two communicating machines are made available through PKI because the identities of the two parties can be verified by way of certificates.

From <https://www.venafi.com/education-center/pki/how-does-pki-work>

3. Certificate Authority
   A Certificate Authority (CA) is used to authenticate the digital identities of the users, which can range from individuals to computer systems to servers. Certificate Authorities prevent falsified entities and manage the life cycle of any given number of digital certificates within the system.

From <https://www.venafi.com/education-center/pki/how-does-pki-work>

4. Registration Authority
   Registration Authority (RA), which is authorized by the Certificate Authority to provide digital certificates to users on a case-by-case basis. All of the certificates that are requested, received, and revoked by both the Certificate Authority and the Registration Authority are stored in an encrypted certificate database.

From <https://www.venafi.com/education-center/pki/how-does-pki-work>

Symmetrical Encryption

Symmetrical encryption protects the single private key

From <https://www.venafi.com/education-center/pki/how-does-pki-work>

Asymmetric Encryption

Asymmetric encryption uses two keys, one public and one private. The public key encrypts and the private key decrypts.

From <https://www.venafi.com/education-center/pki/how-does-pki-work>

API security checklist

Wednesday, March 2, 2022 6:38 PM

API Security Checklist

Checklist of the most important security countermeasures when designing, testing, and releasing your API.

Authentication

- Don't use Basic Auth. Use standard authentication instead (e.g. JWT, OAuth).
- Don't reinvent the wheel in Authentication, token generation, password storage. Use the standards.
- Use Max Retry and jail features in Login.
- Use encryption on all sensitive data.

JWT (JSON Web Token)

- Use a random complicated key (JWT Secret) to make brute forcing the token very hard.
- Don't extract the algorithm from the header. Force the algorithm in the backend (HS256 or RS256).
- Make token expiration (TTL, RTTL) as short as possible.
- Don't store sensitive data in the JWT payload, it can be decoded easily.

OAuth

- Always validate redirect_uri server-side to allow only whitelisted URLs.
- Always try to exchange for code and not tokens (don't allow response_type=token).
- Use state parameter with a random hash to prevent CSRF on the OAuth authentication process.
    - A person that forces(wih malice) a user to perform an action with the server. Like secretly changing mail address
- Define the default scope, and validate scope parameters for each application.

Access

- Limit requests (Throttling) to avoid DDoS / brute-force attacks.
- Use HTTPS on server side to avoid MITM (Man in the Middle Attack).
- Use HSTS header with SSL to avoid SSL Strip attack.
- For private APIs, only allow access from whitelisted IPs/hosts.

Input

- Use the proper HTTP method according to the operation: GET (read), POST (create), PUT/PATCH (replace/update), and DELETE (to delete a record), and respond with 405 Method Not Allowed if the requested method isn't appropriate for the requested resource.
- Validate content-type on request Accept header (Content Negotiation) to allow only your supported format (e.g. application/xml, application/json, etc.) and respond with 406 Not Acceptable response if not matched.
- Validate content-type of posted data as you accept (e.g. application/x-www-form-urlencoded, multipart/form-data, application/json, etc.).
- Validate user input to avoid common vulnerabilities (e.g. XSS, SQL-Injection, Remote Code Execution, etc.).
- Don't use any sensitive data (credentials, Passwords, security tokens, or API keys) in the URL, but use standard Authorization header.
- Use an API Gateway service to enable caching, Rate Limit policies (e.g. Quota, Spike Arrest, or Concurrent Rate Limit) and deploy APIs resources dynamically.

Processing

- Check if all the endpoints are protected behind authentication to avoid broken authentication process.
- User own resource ID should be avoided. Use /me/orders instead of /user/654321/orders.
- Don't auto-increment IDs. Use UUID instead.
- If you are parsing XML files, make sure entity parsing is not enabled to avoid XXE (XML external entity attack).
- If you are parsing XML files, make sure entity expansion is not enabled to avoid Billion Laughs/XML bomb via exponential entity expansion attack.
- Use a CDN for file uploads.
- If you are dealing with huge amount of data, use Workers and Queues to process as much as possible in background and return response fast to avoid HTTP Blocking.
- Do not forget to turn the DEBUG mode OFF.

Output

- Send X-Content-Type-Options: nosniff header.
- Send X-Frame-Options: deny header.
- Send Content-Security-Policy: default-src 'none' header.
- Remove fingerprinting headers - X-Powered-By, Server, X-AspNet-Version, etc.
- Force content-type for your response. If you return application/json, then your content-type response is application/json.
- Don't return sensitive data like credentials, Passwords, or security tokens.
- Return the proper status code according to the operation completed. (e.g. 200 OK, 400 Bad Request, 401 Unauthorized, 405 Method Not Allowed, etc.).

CI & CD

- Audit your design and implementation with unit/integration tests coverage.
- Use a code review process and disregard self-approval.
- Ensure that all components of your services are statically scanned by AV software before pushing to production, including vendor libraries and other dependencies.
- Design a rollback solution for deployments.

From <https://github.com/shieldfy/API-Security-Checklist>

Higher level concepts

Tuesday, October 19, 2021 8:17 PM

5 Most Popular Web App Security Testing Methodologies (apriorit.com)

How to Read and Understand CPU Benchmarks - Intel

https://docs.cyberark.com/Product-Doc/OnlineHelp/PAS/Latest/en/Content/PAS%20INST/PKI-Authentication-Personal-Certificate.htm

https://www.nrc.gov/docs/ML1206/ML12060A141.pdf

https://www.mvorganizing.org/what-is-reconnaissance-in-cyber-security/

Example interview

Wednesday, March 2, 2022 12:47 AM

Talk about your projects and interviews

Technical challenges and complexities

Mvc principles and connecting a lot of APIs together

Mention the impact

Architecture overview

Wednesday, March 2, 2022 6:19 PM

Based on this guide

- https://github.com/donnemartin/system-design-primer#system-design-interview-questions-with-solutions

Useful links

- Study guide
- How to approach a system design interview question
- System design interview questions, **with solutions**
- Object-oriented design interview questions, **with solutions**
- Additional system design interview questions

	Short	Medium	Long
Read through the System design topics to get a broad understanding of how systems work	👍	👍	👍
Read through a few articles in the Company engineering blogs for the companies you are interviewing with	👍	👍	👍
Read through a few Real world architectures	👍	👍	👍
Review How to approach a system design interview question	👍	👍	👍
Work through System design interview questions with solutions	Some	Many	Most
Work through Object-oriented design interview questions with solutions	Some	Many	Most
Review Additional system design interview questions	Some	Many	Most

Step 1: Ask all the use cases, constraints and assumptions
Step 2: Create a high level design
Step 3: design the main components
Step 4: Scale the design

System design video

Vertical scaling

Optimizing processes and increase throughput with same resources. (in restaurant with 1 chef, pay chef more)

Preprocessing and cron job

Preparing resources during nonpeak hours (chef prepping bases during nonpeak times)
Cron job is a scheduling task that is carried out at a certain time

Get a backup
- Avoid a single point of failure. Master(main) chef and slave chef(backup)

Horizontal scaling
- If you get more needs.
- Getting more resources to get more work done.
- (get more chefs and more backups)

Microservices architecture
- Having one set of systems do a certain job very well
- (having a set of pizza chefs and a set of garlic bread chefs)
- Scale differently depending on needs

Distributed system
- Have a whole other environment with sets of resources
- (open another pizza shop)
- Incase there is an outage or a failure of one of the pizza shops
- Partitioning
  - The original pizza shop can delegate orders to the copy pizza shop. Can decide which is more fit to perform the job depending on locality of where the job is coming from
- More fault tolerant and faster to fulfill the request

Load balancer
- Central location to get requests and route them to the most appropriate system. It needs to have realtime specs of how busy and costly the systems will be for time and resources

Decoupling the system
- Having another system have a different set of responsibilities. (the Shops vs the delivery agents)

Logging and metrics
- To make things more efficient for analytics, auditing, reporting, and machine learning

Extensible
- Make things encapsulated so that they can perform the task at hand regardless of inputs (delivery agent doesn't care if it is delivering pizza or burgers. Make it independent of the object being delivered)

High level vs Low level design
- High level is how systems are going to work and interact together
- Lowlevel is writing the code for each of the systems and make classes and objects

Tools

Type	System	Reference(s)
Data processing	MapReduce - Distributed data processing from Google	research.google.com
Data processing	Spark - Distributed data processing from Databricks	slideshare.net
Data processing	Storm - Distributed data processing from Twitter	slideshare.net
Data store	Bigtable - Distributed column-oriented database from Google	harvard.edu
Data store	HBase - Open source implementation of Bigtable	slideshare.net
Data store	Cassandra - Distributed column-oriented database from Facebook	slideshare.net

Data store	DynamoDB - Document-oriented database from Amazon	harvard.edu
Data store	MongoDB - Document-oriented database	slideshare.net
Data store	Spanner - Globally-distributed database from Google	research.google.com
Data store	Memcached - Distributed memory caching system	slideshare.net
Data store	Redis - Distributed memory caching system with persistence and value types	slideshare.net
File system	Google File System (GFS) - Distributed file system	research.google.com
File system	Hadoop File System (HDFS) - Open source implementation of GFS	apache.org
Misc	Chubby - Lock service for loosely-coupled distributed systems from Google	research.google.com
Misc	Dapper - Distributed systems tracing infrastructure	research.google.com
Misc	Kafka - Pub/sub message queue from LinkedIn	slideshare.net
Misc	Zookeeper - Centralized infrastructure and services enabling synchronization	slideshare.net

From <https://github.com/donnemartin/system-design-primer#real-world-architectures>

- Clones
  - All of the servers should return the same things. The load balancer will decide which server will hand the task but regardless the response should be the same. And in the case that you need to horizontally scale your system, you should be able to create an exact copy as your other servers and it should work along with the entire system.
- Database
  - MySQL doesn't scale well because of it being a relational database. Use a NoSQL from the start like MongoDB and denormalize from the beginning. But eventually you will still need a cache to keep up with size of demand.
- Caching
  - DON'T DO FILE BASED CACHING since it is a pain for cloning and autoscaling
  - Holds the dataset in RAM so it is super fast
  - Cached database queries
    - Store query results and don't need to query again
    - But it expires quick since database is constantly changing values
  - Cache objects
    - Store a dataset as an instance
    - Change the dataset as necessary
    - Async possible
- Asynchronism
  - One method of async is to have a preprocessing cronjob to compute before hand of needed objects
  - To have the front end still going as the background is doing a job
  - Learn from RabbitMQ in order to learn async processes

- Scalability
    - Scalable if we can increase the resources in a system to increase performance
    - Redundancy should not slow performance
    - Heterogeneity meaning there is a lot of diverse systems, new/old, faster/slower systems.
        - Algorithm of uniformity will break down or underutilize systems
    - Throughput vs latency
        - Throughput is how much data at once vs latency is amount of time it takes data to arrive
        - What you probably want is maximum throughput with acceptable latency
    - Failover
        - What to do during a fail
    - Master slave
        - Master can do reads and writes and the slave can do reads.
    - Buddy replication
        - Circular backup for a system A backup for B backup for C backup for A

RDBMS vs NoSQL

- CAP (consistency, availability, partitioning tolerance)
    - Consistency
        - Every read will get the most recent write (or error)
    - Availability
        - Every request will get a response. Regardless of validity of data
    - Partition tolerance
        - Continues to operate despite a couple resources go down
    - AP
        - Data might not be the latest but eventually it will be right
    - CP
        - Might get errors but good for atomic reads and writes

Benchmark testing

Sunday, March 6, 2022 9:11 PM

4 popular testing libraries in Python

- Timeit

  A **Benchmark in Performance Testing** is a metric or a point of reference against which software products or services can be compared to assess the quality measures. In other words, Benchmark means a set standard that helps to determine the quality of software product or service.

  From <https://www.guru99.com/benchmark-testing.html>

- Line_profiler

  The line_profiler library allows you to get the execution time of each individual line in a file. This is incredibly useful if you're having trouble narrowing down slow functions or third-party calls in a larger file

  From <https://medium.com/swlh/4-simple-libraries-to-quickly-benchmark-python-code-8d3dfd288d7a>

- Resource

  Setting unbounded processes off to chew through cycles is taboo and could land you in a world of pain. That's where resource comes in. This library will let you measure resource usage in your code and even set limitations on how much of a particular resource can be consumed.

  From <https://medium.com/swlh/4-simple-libraries-to-quickly-benchmark-python-code-8d3dfd288d7a>

- Memory_profiler

  Setting unbounded processes off to chew through cycles is taboo and could land you in a world of pain. That's where resource comes in. This library will let you measure resource usage in your code and even set limitations on how much of a particular resource can be consumed.

  From <https://medium.com/swlh/4-simple-libraries-to-quickly-benchmark-python-code-8d3dfd288d7a>

Chat app (system design example)

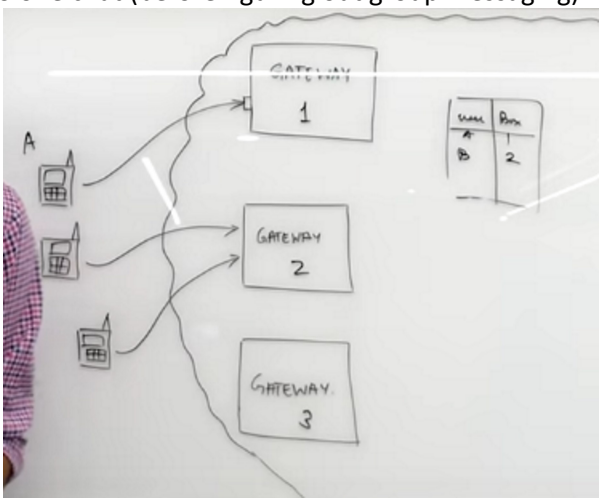
Thursday, April 7, 2022 7:14 PM

Ask questions about features

- Usually they will agree to the first one
    - So ask for something simple and something you already know
- The ones in this example
    - Group messaging
    - Sent + delivered + read Receipts
    - Online/last seen
    - Image sharing
    - Chats are temporary or permanent

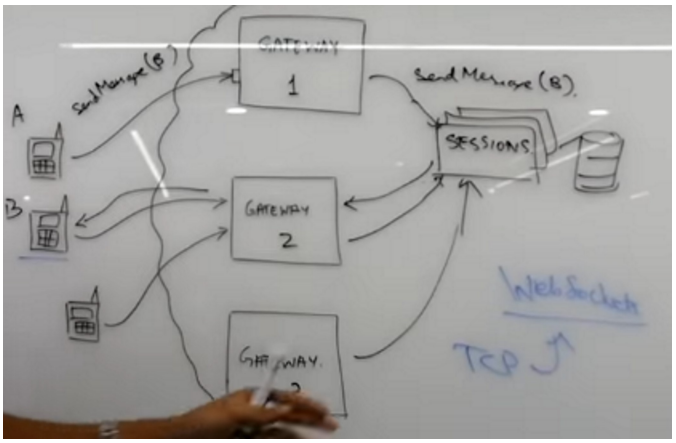
One to one chat (before figuring out group messaging)

-

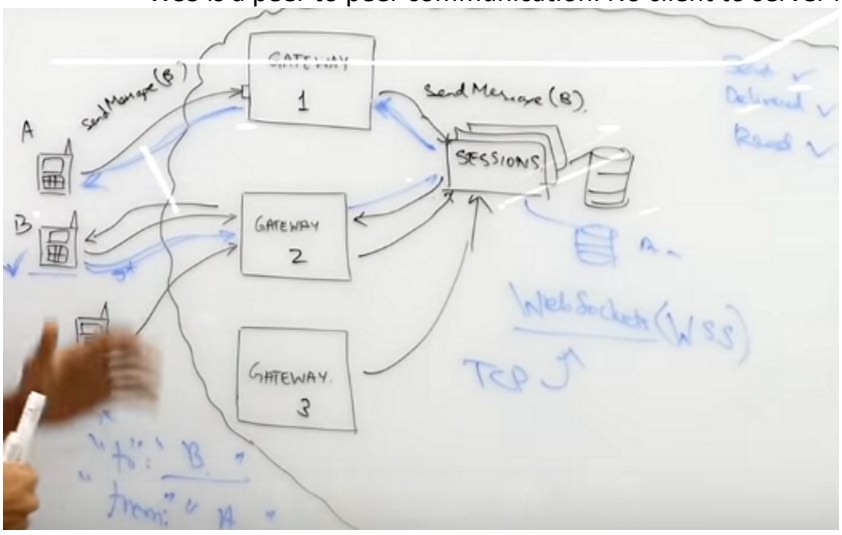


    - A and B need to know where eachother are.
    - Store the box (gateway) to user mapping somewhere else
    - We want to be efficient with space on the boxes. TCP connections take up some memory
    - The box to user info would be repeated on all the boxes. So might as well have a central location
    - Make it a microservice that store session info. Multiples of them to avoid single point of failure.
    - Sessions service acts as a router

-

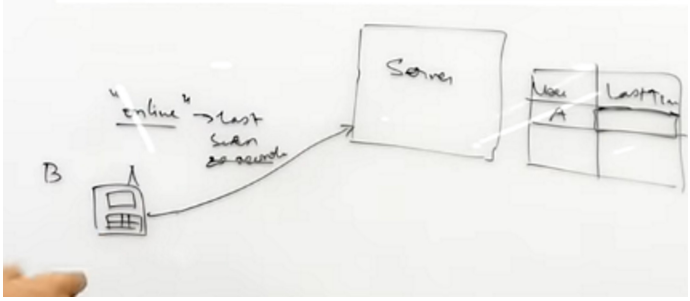


    - We need the client to send a message to server and the other server to send a message to the receiving client.
    - We might be able to use HTTP if we use long polling but that makes it not real time

    - So lets use WEBSOCKETS
        - WSS is a peer to peer communication. No client to server hierarchy
- 

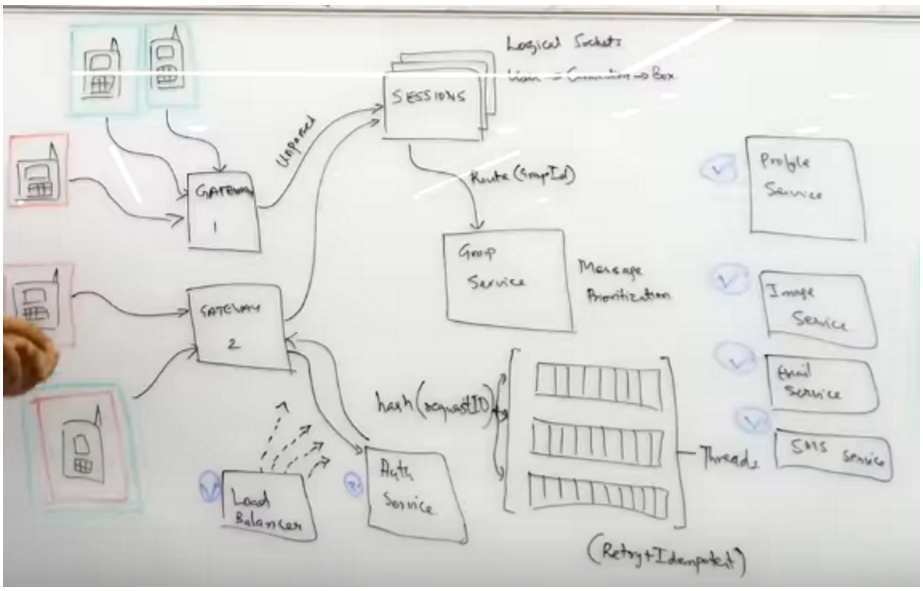
    - For sent, have a database on the sessions end that will return a sent.
    - Once the receiver gets the message, have it send back a delivered to the sender
    - And once the receiver sees the message, send a seen acknowledgement to the sender

Last seen time stamp

- We need to choose which requests and functions will trigger online activity
- 

    - Last seen microservice is the server
    - Delivery will not trigger the user activity
    - App request vs user activity
        - Don’t send the update to the last seen microservice if it is an apprequest as opposed to a user activity
    - Need a buffer of last seen time (3 seconds ago is too recent)

Group messaging

- 

- Session stores info of all groups and its too complicated
- Group service connected to the sessions service
    - Group service can respond with whose all in a group
- Sessions will take all the members of the group and then send the message
- Limit the number of people in a group to have sessions have a normal amount of users per sessions microservices
- Have a parser microservice
    - To analyze the message that has not been parsed
    - Send the right messages to the right place from the parser
- Consistent hashing
    - Some informations to some boxes
    - Route request to the right box
- Message queues gaurantees that the message will be sent and a failure will notify the client
- Deprioritize messages
    - During surges who cares if delivered before the seen

System design crash course Guarav

Thursday, April 7, 2022 10:01 PM

Consistent hashing

- We want to load balance
- Uniform load to all servers based on hashing
- When adding server, all of the other users have to reallocate where the requests go to.
- Need to have advanced approaches in order to reallocate efficiently.
- Put all the servers and requests in a circle
- Add into a blank entry to add server
- Add virtual servers in order to split the load if one goes down

Message queue

- Not the pizza but a confirmation that the order has been placed.
- Adding to the queue of making pizza orders
- The main point is that the process was asynchronous
- Priority can be put in place and have a priority queue
- Need a database in order to store the queue so that if the pizza store goes down then other ones can take care of it
- If the store goes down, then other stores have to cover and clients get rerouted to different server.
- Through load balancing and heart beat mechanism, let the failed orders go to the new server
- Message Queue has all the following
    - load balancing
    - Heart beat
    - Assignment
    - Persistence

Microservices

- Versus monolith
- Can still horizontally scale with the monolith
- Microservices usually talk to their own databases
- Monolith
    - Good for small teams
    - May not have the time and resources to make microservices
    - Less complex
    - Faster procedural calls
    - New members have a lot more context to learn
    - Deployments are complicated since it affects the entire systems
    - Too much responsibilities on each servers with no separation
- Microservice
    - Easier to scale
    - Dev has less to learn for just the server
    - Parallel dev is easy since less dependency
    - Less coupling so easier upgrade easier
    - Needs to skilled architect for all moving parts

Data Sharding

- Horizontal partitioning - split up the data and put them on different databaseservers. Split up by and ID

- Vertical partitioning is splitting up columns of data
- Consistency is more important than availability in most cases
- Horizontal partitioning have a hard time joining across shards
    - Need multiple queries then join
- Indexing in shards are good
- Master slave architecture for every shard
- Consistency is tough for sharding

Distributed Caching

- Caching
    - In memory
    - Key and value
    - Save network calls to database
    - Avoid computations by storing metadata calculations like average age of all entries
- Cant store everything
    - More stuff in cache means more search time
- Needs most relevant information
    - Predict the needed data
- Policy load and evict data from cache
- LRU - least recently used get kicked out
- Harmful to have a poor policy since you are making calls to unneeded calls to cache if its not going to be worth it.
- If cache too small then its called thrashing
    - Constantly inputting and overwriting without using cache
- Updated entries needs to be reflected in cache
- Global cache for all servers. Midpoint between database and server
    - LIKE REDIS
    - A little slower but more accurate
- Write through
    - Update the entry in cache then push to database
    - What if there are other servers with same cached info that needs the update
- Write back
    - Hit database and then update the entry in the cache
    - Is expensive and a little slower
- Use a hybrid approach
    - Write to the cache
    - Bulk writes to update the database once a bunch of entries in the cache have been updated
    - Works for only for non critical information

EDUCATIVE.com for system design concepts and examples

Netflix adding videos

Tuesday, April 12, 2022 11:50 PM

Need multiple formats because people have different internet speeds.

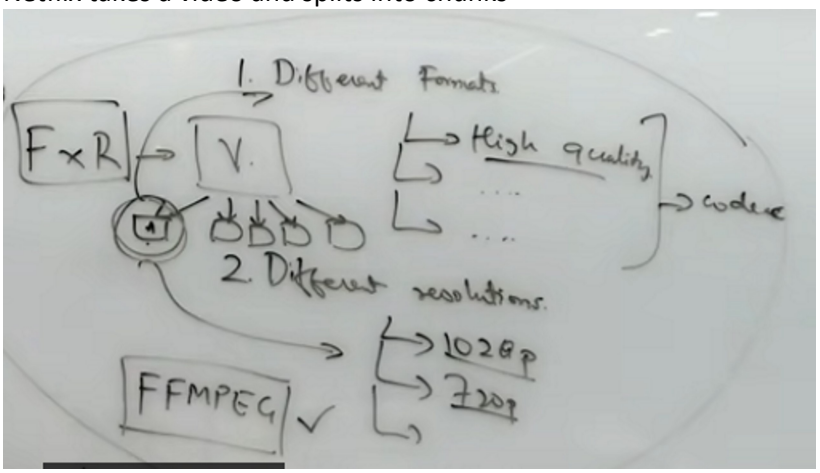
- High qualities
- Streaming speeds
- Codec is the way you compress videos

Different resolutions

- Cellphone vs tv or laptop

Creating touples of qualities and resolutions

Netflix takes a video and splits into chunks



And each chunk has a different format and resolution

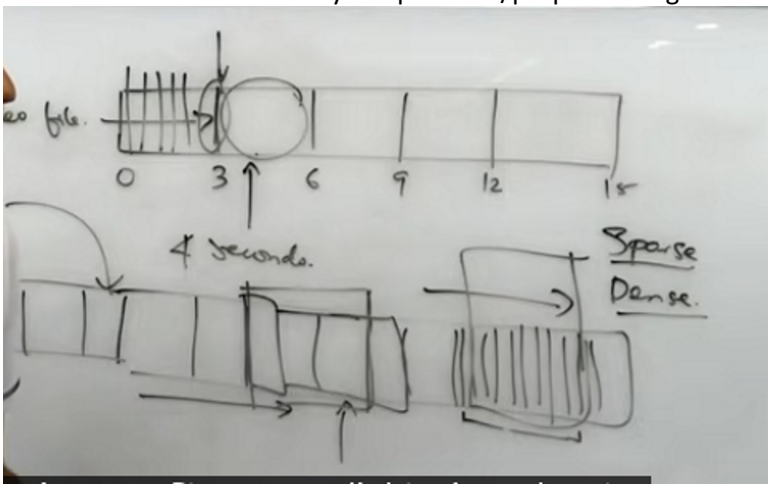
Used to be 3 minutes each
Break the timestamps based on SCENES

Based on user habits it can be sparse where the user is clicking through

- Just get that certain scene that is being skipped to

Vs dense where the movie is being watched the entire way through

- Fetch future scenes as a way of optimized/preprocessing

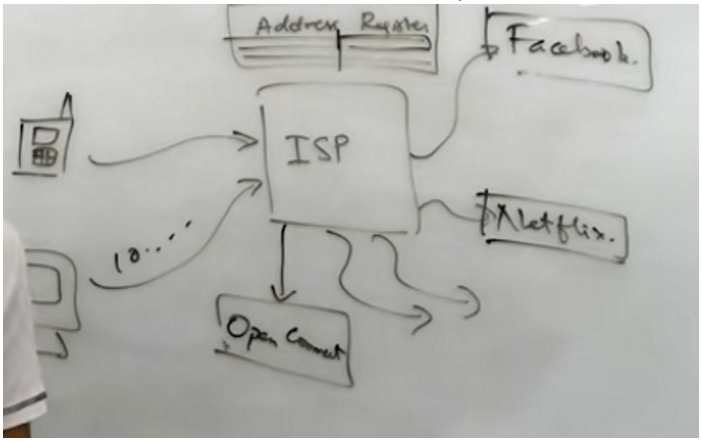


Amazon s3

- storing static data. Something that is not changing and its cheap

To improve on user experience

- Cache
    - Precompute and store it
- ISP calls open connect
    - Which is a cache of movies
    - Great for international request where the data is all housed in the US



- Bandwith saved
- Reduce load on ISP
- Local popular movies
- Update the open connect box to update the newest hot movie

Tindr system design

Wednesday, April 13, 2022 10:23 PM

Store profiles
- Will have images for profiles
- Example of 5 images per user

Recommend matches
- How many active users? Is a good question

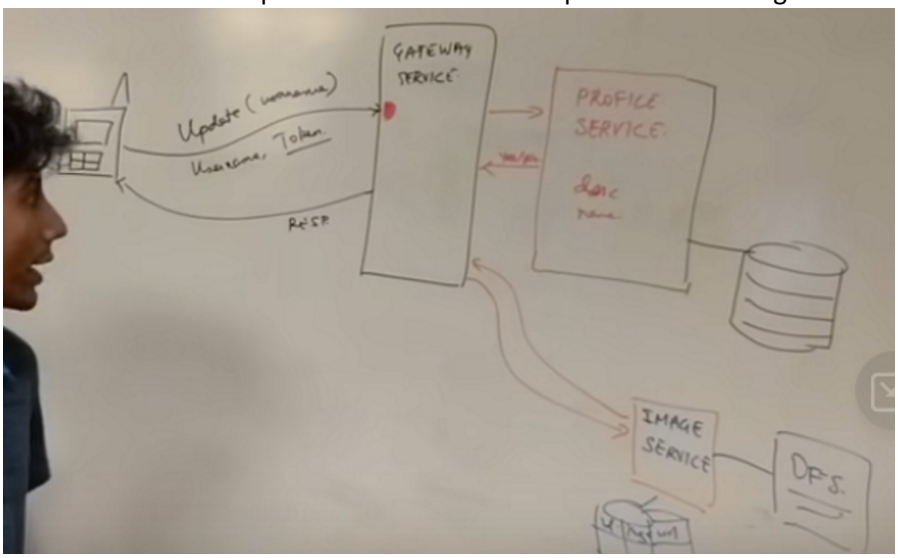
Note matches
- Percentage of matches
- 0.1% per swipe

Direct messaging
-

Two approaches
- Start with ER diagram
    - Too constrained and abstract for user needs
    - How data will be modeled
    - Back to front thinking
- Go from front to back
    - Think about what your users need
    - Think about how services will be broken down
    - Then think about data requirements
    - Feature development first

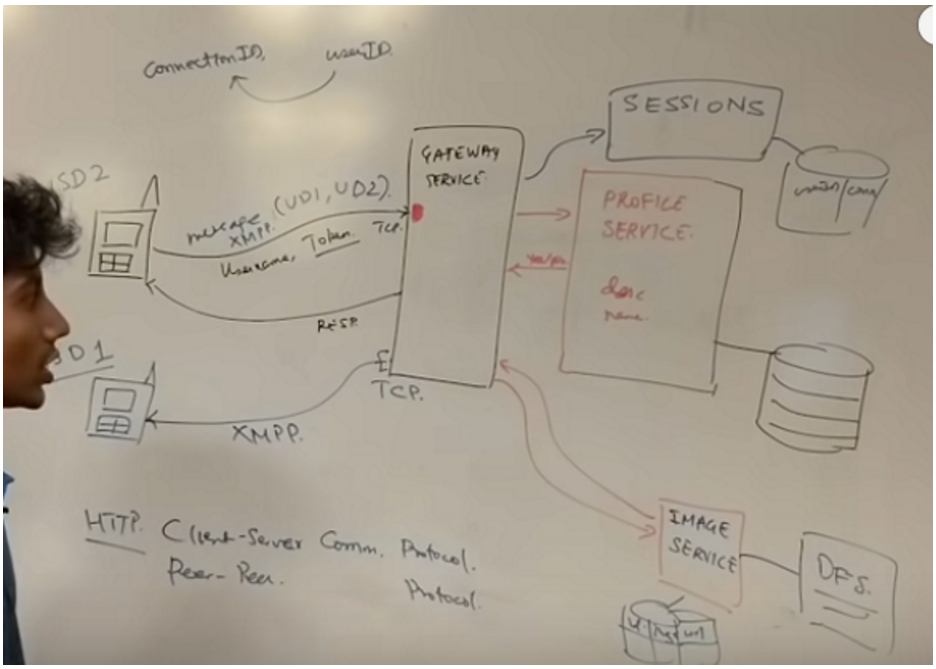
Storing images
- How are you going to store images
    - File or blob
    - Blob = binary large object
    - Guarantees from database
        - Mutability(don't need changes just replace)
        - Transaction(don’t need images to be atomic)
        - Indexes (good for search, but don’t need to search for contents of file)
        - Access control(can get the same file system controls)
    - Storing files
        - Cheaper
        - Faster(maybe)
        - Static content delivery network (CDN)

Storing profiles
- Profiles service with database
- Maybe connected to email service with 2 step authentication
- Update profile
    - Need to make sure the command comes from the right user
    - Send user name and token instead of username and password because security
- Use a gateway service
    - Only one to talk to the client
    - Takes request, asks profile service if the user is authenticated
    - Directs the request to the proper service
- Image service
    - For heavy computations needing all images

    - Maybe all the details of the profile like age is a separate service
    - Have a distributed file system of the images
    - Have a separate database that has profile id and image id
- 

Direct messages

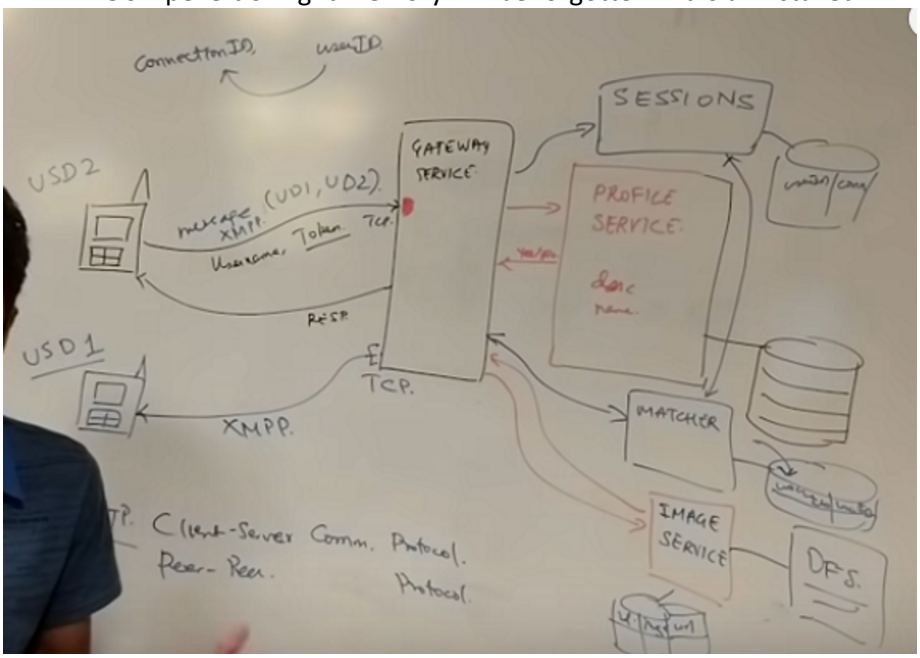
- Cannot have chat with http because of client is always the initiator
- Need to use a peer to peer protocol
    - Can use XMPP
- Web socket connection or maybe a custom TCP
- Who will maintain the connection info
    - We want to decouple the systems as much as possible
    - Another service that can handle sessions
    - Userid to connection id info



Noting matches

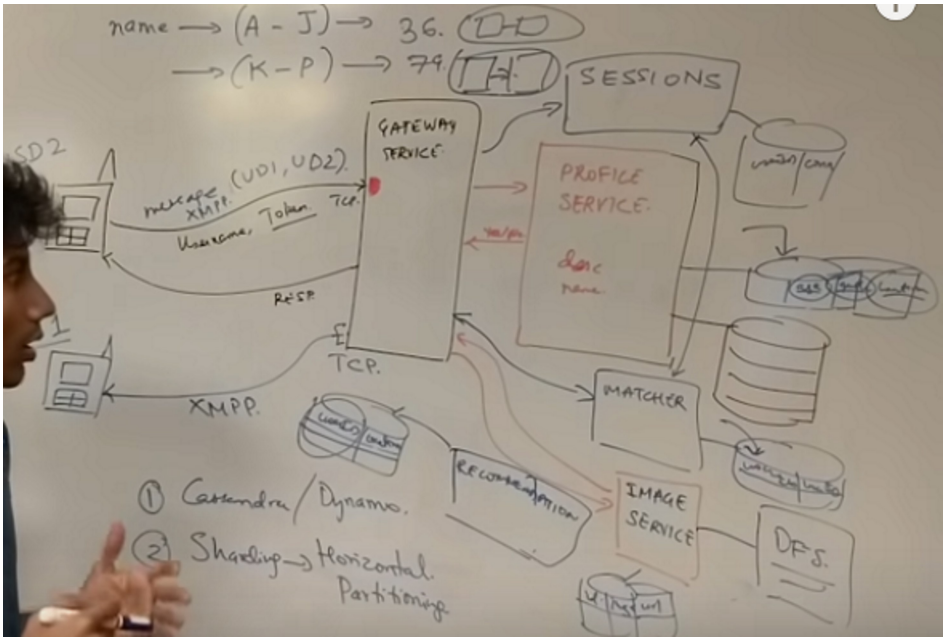
- Where to store the matches
- Matches service

    - Connections to the sessions service to have conversation after the match
- Store a database of userid to userid of matches
- The swipe left or right memory will be forgotten if it is uninstalled



Recommendation service

- Which person is close to me
- Profile service will have a database of age, gender and location
- It is not possible to index on multiple columns in the table
    - So maybe a good database to use would be cassandra which is a Nosql (nonrelational/distributed) database
    - Could use sharding (horizontal partitioning) on relational database
- Recommendation service
    - Pulls all relevant people
    - Stores userid and location



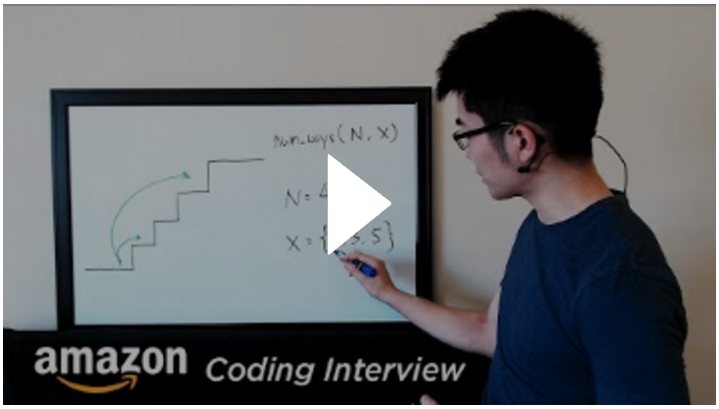
Low level design

Wednesday, April 20, 2022 11:21 PM

Example 1

Tuesday, March 1, 2022 7:07 PM

[Amazon Coding Interview Question - Recursive Staircase Problem]



Basics: can only take steps from a certain set (if the set is {1,2} then you can only take 2 steps). Reach the top and count how many combinations of steps can be take to reach the top. How many possibilities?

Initial Idea: Tree (only where)
Set_values
Final_steps_set
Child_nodes_set
Current_traversals_steps
Next_traversals_steps
For values_so_far, steps_left in Current_traversals_steps:
    For set_value in set_values:
        If set_value == steps_left:
            Final_steps_set.add((values_so_far.concat(set_value))
        elIf set_value < steps_left:
            Next_traversals_steps.add(values_so_far.concat(set_value), steps_left-set_value)
        Else:
            Continue
Current_traversals = next_traversals

While true:
    If current_traversals.size > 0
        Call function
    Else:
        Return sizeof(final_steps_set)

Actual solution:
N=number of steps
X = set of steps that can be take
Num_ways(n, x)
If n==0 : return 1

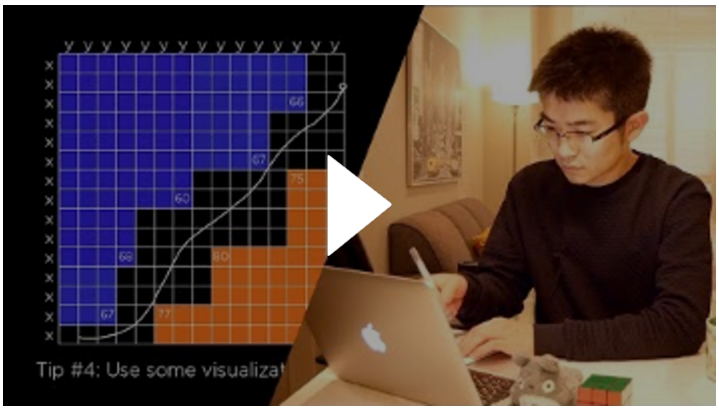
```
Nums = new int[n+1]
Nums[0] = 1
For i in range(1, n):
        Total = 0
        For j in x:
                If i-j >= 0:
                Total += nums[i-j]
        Nums[i] = total
Return nums[n]
```

What did I learn:
Don’t verbosely do things. If it is how many different ways, then try to work with numbers. Use numbers of previously calculated values in order to save on efficiency

Tips how to solve

Tuesday, March 1, 2022 8:01 PM

[5 Problem Solving Tips for Cracking Coding Interview Questions]



1. Come up with a brute force method
2. Think of simpler version of the problem
3. Think with simpler examples
    a. Go through if there is a convenient inputs
4. Use some visualizations
5. Test your code on a few examples

Top algorithms to know

- Tree traversals
    - Preorder traversal
    - In order traversals
    - Postorder traversals
    - Breath first search
    - Depth first search
- Know data structures to represent a tree
    - Can have a node and then an array with children nodes as elements
- Recursion is hard and don't know it elegantly
    - If youre going to do recursion, then have an outer function that does the initializations and then have helper functions that use them
    - Always find the base case first in a recursive function
    - Recursion is not used often because it is heavy on stack space
    - Convert that into iterative function using stacks or

queues
- Stacks and queues
    - If they are balanced then pop and push queues
    - Know them
- OOP
    - Bring it up wherever you can
    - Organize code
    - Know
        - Create class
        - Setup methods
        - Private variables
        - Public variables
        - When to build classes
            - May actually save time because of edge cases. Progress faster
- Hashmaps
    - Simple trick
    - If youre stuck then bring up hash maps, stack, queues
    - How to create and populate hashmaps
        - What is a hash function and give example
        - Maybe something like the sum of the array values LEARN THIS
- How many passes and count as a linear time algorithm
- C or c++ to do things like reverse a linked list
    - Because they have pointers
    - Interviewers might ask that just to make sure
- Sorting
    - Runtime analysis of it
    - They probably wont ask you hot to implement them
- How to work with strings in your chosen language
    - Stuff like making sure that the string is a palindrome
- Dynamic programming
    - Aha moment where it is a stroke of genius
    - Know the basics but don’t spend too much time
    - Usually memorization and caching the values
        - Using sub problems and solving the larger ones
        - Using previously calculated stuff
- Solving the problem is not the goal. How well can you analyze and how well can you look at tradeoff of time and memory.
- Space efficient but not time efficient and point out the pros and cons

Coding practice

Wednesday, March 2, 2022 12:50 AM

Traversing binary trees and simple and common things

Leetcode

Questions

Passing in a function and returning a function
Binary search and log2 time you can find which version is bad
Should take 0 space

Dynamic programming and save an entire array of sums instead of having to compute every single time

Depth first search for loop detection

Sorting based on key

```
lst =[('Mark',1),('Jack',5),('Jake',7),('Sam',3)]
lst_sorted =sorted(lst,key=lambdax:x[1])
```

Prefix sum

- A sum of everything so far in the array at that index

Object Oriented design

Wednesday, March 2, 2022 6:30 PM

More examples

Tuesday, April 5, 2022 12:39 AM

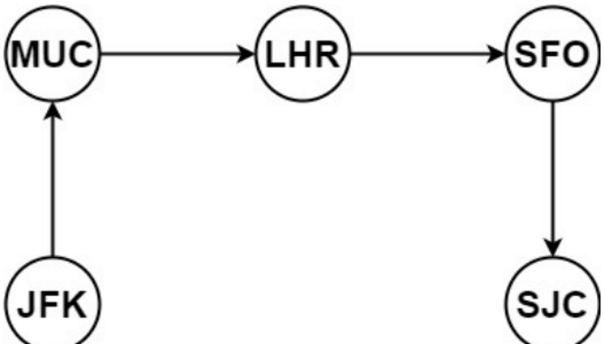
You are given a list of airline `tickets` where `tickets[i] = [from_i, to_i]` represent the departure and the arrival airports of one flight. Reconstruct the itinerary in order and return it.

All of the tickets belong to a man who departs from `"JFK"`, thus, the itinerary must begin with `"JFK"`. If there are multiple valid itineraries, you should return the itinerary that has the smallest lexical order when read as a single string.

- For example, the itinerary `["JFK", "LGA"]` has a smaller lexical order than `["JFK", "LGB"]`.

You may assume all tickets form at least one valid itinerary. You must use all the tickets once and only once.

Example 1:



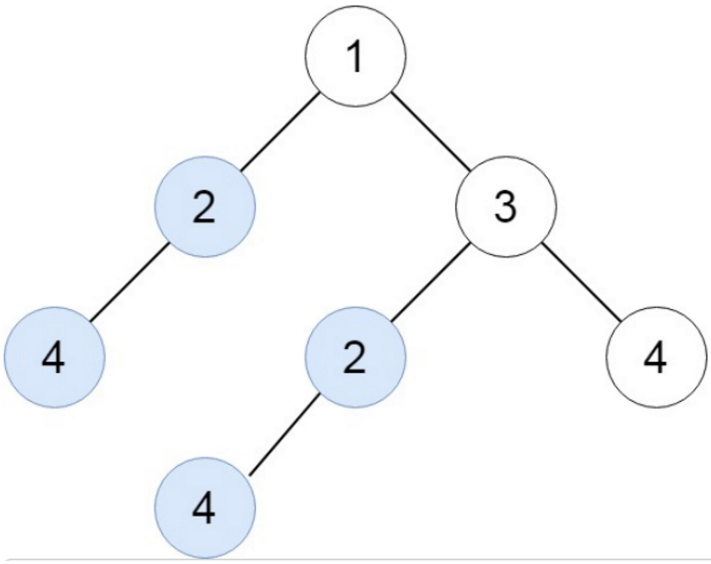
Need to take into account backtracking. If there is another flight from the same location, make sure you can make your way back to the airport before taking that path.

Given the `root` of a binary tree, return all **duplicate subtrees**.

For each kind of duplicate subtrees, you only need to return the root node of any **one** of them.

Two trees are **duplicate** if they have the **same structure** with the **same node values**.

Example 1:



Use recursion to go to the end of the tree and find if any subtrees are the same

- Given `n` non-negative integers representing an elevation map where the width of each bar is `1`, compute how much water it can trap after raining.

  Example 1:


  ```
  Input: height = [0,1,0,2,1,0,1,3,2,1,2,1]
  Output: 6
  Explanation: The above elevation map (black section) is represented by array [0,1,0,2,1,0,1,3,2,1,2,1]. In this c
  ```

- Use stack to store the indices of the bars.
- Iterate the array:
  - While stack is not empty and height[current]> height[st.top()]\text{height[current]}>\text{height[st.top()]}height[current]>height[st.top()]
    - It means that the stack element can be popped. Pop the top element as top\text{top}top.
    - Find the distance between the current element and the element at top of stack, which is to

be filled. distance=current−st.top()−1\text{distance} = \text{current} - \text{st.top}() - 1distance=current−st.top()−1
    - Find the bounded height bounded_height=min (height[current],height[st.top()])−height[top]\text{bounded_height} = \min(\text{height[current]}, \text{height[st.top()]}) - \text{height[top]} bounded_height=min(height[current],height[st.top()])−height[top]
    - Add resulting trapped water to answer ans+=distance×bounded_height\text{ans} \mathrel{+}= \text{distance} \times \text{bounded_height}ans+=distance×bounded_height
- Push current index to top of the stack
- Move current\text{current}currentto the next position

Given an integer array `nums` of **unique** elements, return *all possible subsets (the power set)*.

The solution set **must not** contain duplicate subsets. Return the solution in **any order**.

Example 1:

```
Input: nums = [1,2,3]
Output: [[],[1],[2],[1,2],[3],[1,3],[2,3],[1,2,3]]
```

Example 2:

```
Input: nums = [0]
Output: [[],[0]]
```

When taking into account neighbors. Think about having to iterate twice. Once from forwards and once backwards

Defaultdict(default init val type)
Dictionary.items will give you both key value pair

```
class Solution:
    def kClosest(self, points: List[List[int]], k: int) -> List[List[int]]:
        # Sort the list with a custom comparator function
        points.sort(key=self.squared_distance)

        # Return the first k elements of the sorted list
        return points[:k]
```

```
    def squared_distance(self, point: List[int]) -> int:
        """Calculate and return the squared Euclidean distance."""
        return point[0] ** 2 + point[1] ** 2



"""Robbing a house where the adjacent houses cannot be robbed. Get the maximum amount"
class Solution:
    def rob(self, nums: List[int]) -> int:
        n = len(nums)
        if n == 1:
            return nums[0]
        if n == 2:
            return max(nums[0], nums[1])

        n2 = 0
        n1 = nums[-1]
        curr = 0

        for i in reversed(range(n-1)):
            curr = max(n1, n2 + nums[i])
            n2 = n1
            n1 = curr

        return n1
```

BFS and DFS

Tuesday, April 5, 2022 5:25 PM

```
visited =[] # List for visited nodes.
queue =[]    #Initialize a queue
Def bfs(visited, graph, node): #function for BFS
  visited.append(node)
  queue.append(node)
While queue:          # Creating loop to visit each node
    m =queue.pop(0)
    print(m, end =" ")
    for neighbour in graph[m]:
      if neighbour not in visited:
        visited.append(neighbour)
        queue.append(neighbour)
```

From <https://favtutor.com/blogs/breadth-first-search-python>

```
Graph = [initgraph]

visited =set() # Set to keep track of visited nodes of graph.
Def dfs(visited, graph, node):  #function for dfs
        ifnode not in visited:
                print(node)
                visited.add(node)
                for neighbour in graph[node]:
                   dfs(visited, graph, neighbour)
```

From <https://favtutor.com/blogs/depth-first-search-python>

Data structures

Wednesday, April 6, 2022 9:26 AM

- Just a regular list:
  - Stack
  - Queue
    - Both stack and queue just use the pop and push methods of a list
    - Can be implemented using deque (more efficient)
- Regular list with particular library
  - Heap
    - Using heapq
    - Heap is a list ordered from lowest to highest
- It's own data type
  - Double ended queue
    - Deque library
  - DefaultDict
    - Ordinary dictionary {}
  - OrderedDict
    - orderedDict library
  - Counter
    - Counts the occurrence in a list

- Depth first search for loop detection

Useful functions

Monday, June 6, 2022 10:14 PM

Python

- Array.pop(0)
    - will get the start of the array and take it out of the array
- Sorted(array)
    - Will sort array and return new array from least to greatest
- Array.sort()
    - Will sort the current array with no output

Crash Course

Monday, April 18, 2022 8:06 PM

Data Wrangling
- Cleaning
- Structuring
- Enriching data
- For better decision making in less time

Models
- Systems that learn by finding and applying patterns from previous observations
- Training = model development
- Never test your model on the same data used to train it
- Data leakage
    - Information from outside the training set somehow gets in
    - If the test data set gets into the training set
- Overfitting
    - Not a good predictor since the data doesn't vary much

Data
- Data split I usually 80 20 training and testing
- Variables
    - Parameters
        - Variables whos values are determined during the training
    - Hyperparameters
        - Variable values determined before the training
        - For example, sum of a couple of relevant columns
    - Explanatory variables
        - Independent, features, predictors
    - Target variable
        - Dependent variable

Training
- Supervised learning
    - Clearly defined input with clear output
    - Regression
        - Relationship between independent variables and continuous dependent variables
    - Classification
        - Relationship between independent variables and categorical dependent variables
- Unsupervised
    - Defined input with unclear output
        - Like a clustering algorithm to classify based on nondecided classification outputs
- Semisupervised learning
    - In between
    - Like labeled input and output will infer the missing labels
    - Retrain until all of the data is labeled

Performance
- Perfromance metrics

    - Accuracy - percent guessed correctly
    - Sometimes not good like when guessing pricing despite it being a good model

Analysis
- Mean squared error analysis
    - For regression analysis
    - Good to find pricing off by how much

Scores
- F1 score for imbalanced data sets
    - True positive
    - True negative
    - False positive
    - False negative
    - F1 Score formula
        - Precision: $\frac{TP}{TP+FP}$
        - Recall: $\frac{TP}{TP+FN}$
        - $$F_1 = \frac{2 \times precision \times recall}{precision + recall}$$
- F1 score used for binary classification
    - All the F1, precision, and recall all fall into 0 to 1
    - CLOSER TO 1 THE BETTER

THE HARD PART OF DATA SCIENCE

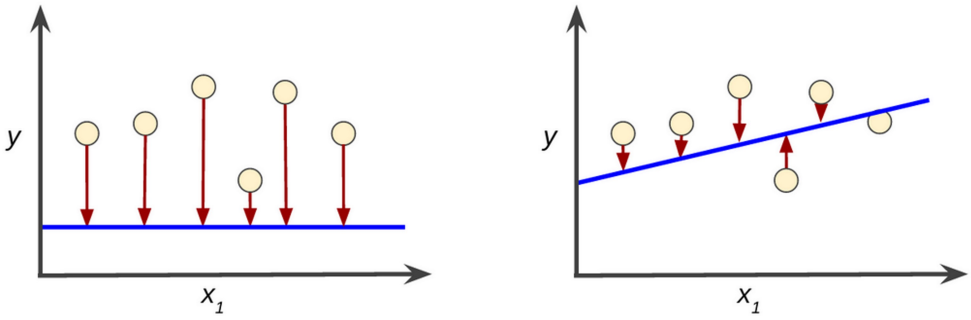
- quantifying a business problem as a Data Science problem
- setting up the framework for data ingestion
- engineering your features for better representation
- understanding and communicating reasonable expectations and limitations of your work
- applying model inferences as a part of a larger business system
- automating model re-training as you obtain more data

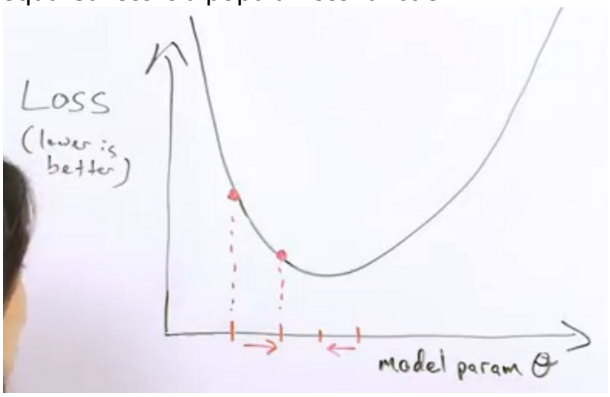
^ is a hat that means estimation

Feature Engineering
- Steps to raw data before training

Loss
- Loss is the penalty for a bad prediction. That is, **loss** is a number indicating how bad the model's prediction was on a single example. If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater.

- The arrows represent loss.
- The blue lines represent predictions.



- Squared loss is a popular loss function
- 

- Learning rate is how big of a difference to move over to get closer to lowest loss. Too little and it would take too many iterations to get there. Too large and you would jump all over the place
- Batch size is all the data points of the loss calculation. This becomes a problem when there are millions of datasets and you don't want to have to generate a graph from all of the loss values with millions of points
- It will always end up with a bowl convex graph
- Gradient decent is getting the slope and going towards the lower
- Stochastic gradient decent is choosing a random sample and going lower based on that one sample (batch size of one)
- Mini-batch stochastic gradient has anywhere between 100 to 1000 samples as the convex graph.

Data representation and cleaning

Monday, April 18, 2022 10:47 PM

Continuous variables REGRESSION

- Like income
- Can have groups of income to make it a categorical variable

Categorical variables CLASSIFICATION

- Finite set of possible values
- One hot encoding
    - Making the categories into number values
    - No payment : 0, credit card: 1, Paypal : 2
- Ordinal
    - Categorical values that can be ordered

Interaction between variables

- Sometimes good to explicitly note some difference like a 20 year old with a 150k salary vs a 50 year old with a 150k salary
    - Can multiply age and salary to get an interaction variable
- Categorical interaction
    - Get essentially a table of possibilities
- Continuous and categorical interaction

Data cleaning

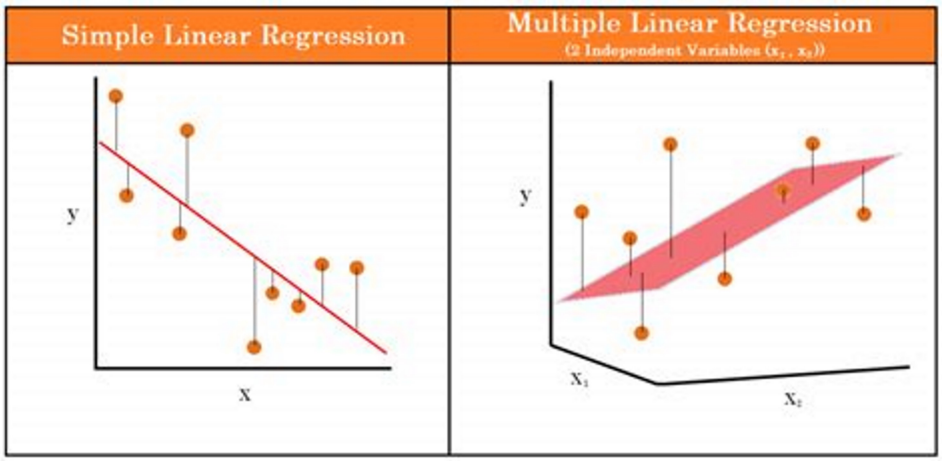
Cleaning

- Process of correcting or removing inaccurate records
- Normalization
    - Combine results such as software engineer and developer title
    - Make between 1 and zero
- Real data is messy, biased, and potentially inaccurate
- Ways to identify if data is wrong.
    - If people filled out a survey in mere seconds and didn't read the survey questions
- Correct data if they are conflicting like being employed with no income

Common ML Algorithms

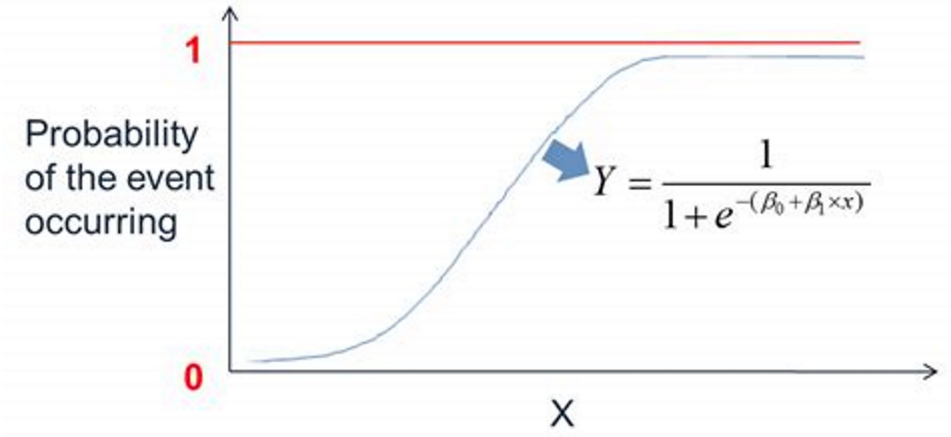
Tuesday, April 19, 2022 7:41 PM

Linear regression

- A continuous dependent variable (lines)
- Simple regression
  - One dependent variable
- Multiple linear regression
  - Multiple dependent variables
- 

- error term known as residuals
  - The difference between the line and the actual data points
- This doesn't work when the independent variables are highly correlated
- TOO SIMPLE sometimes
  - Very easy to understand because of it but it might be too simple to tackle real world problems

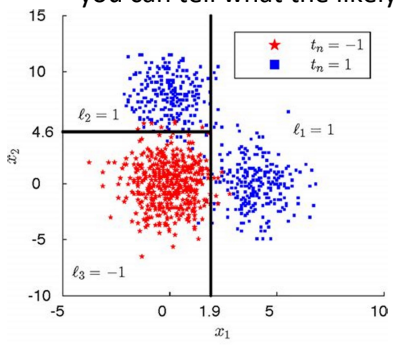
Logistic regression

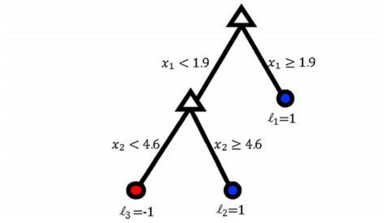
- Probability that something will happen
  - Or a certain choice will be made
- 

- It is good for when the outcome is in two categories of yes or no
- Can be used for classification of 2 different groups

Decision trees

- For both classification and regression
- Divide plots into regions
    - If your new plot lands in a region, then you can set the average price for that region
    - Same for regression and classification if your new plot point lands on a certain region then you can tell what the likely outcome will be.
-



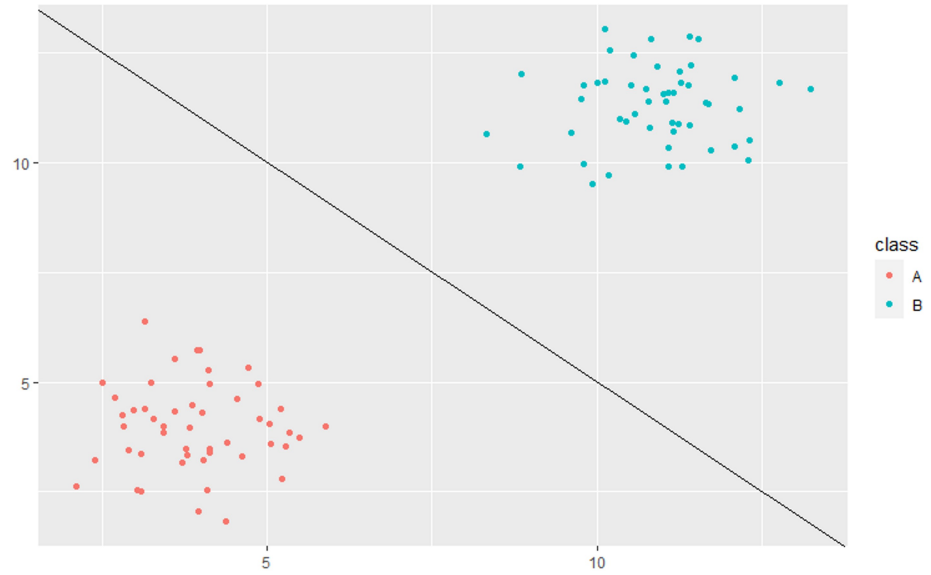


- Bagging
    - Decision tree for subsets of data
    - The decision is then the average of all the subsets
- Boosting
    - Learning sequentially from previous decision trees
- Random forest
    - Uses bagging
    - Random samples for each tree

Support vector machines

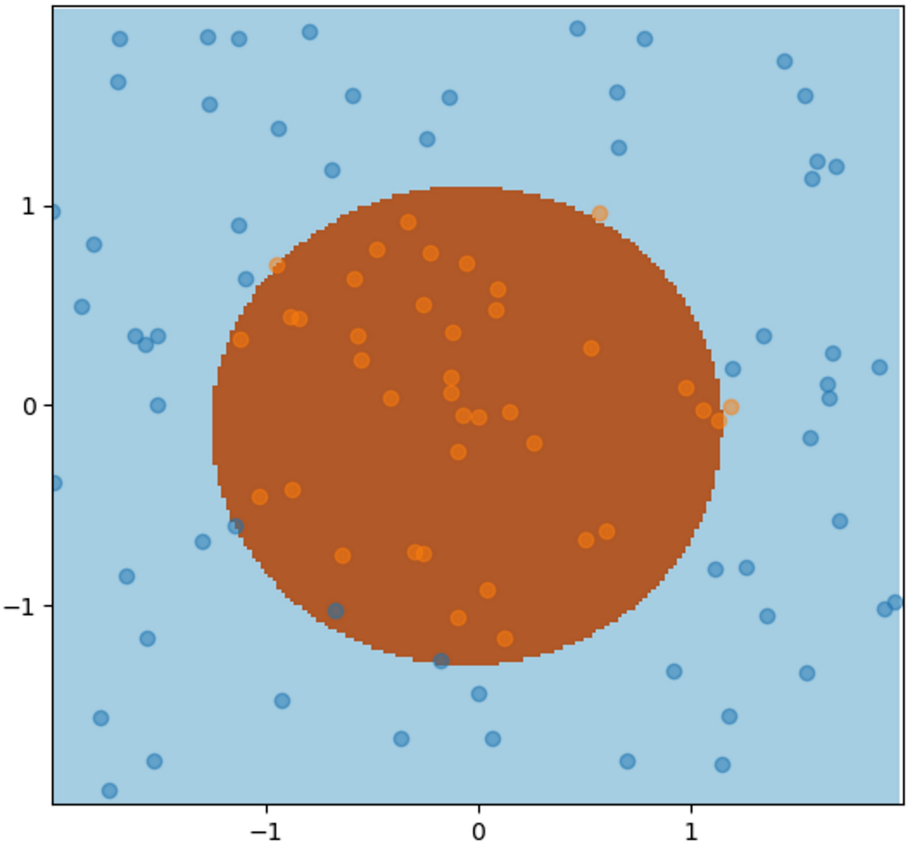
- Classification based on graph location
- Linear separation

-



- Can have hard or soft margins(separator)
    - Soft margins occur when there is no clear cut line between the classification
    - Hard margins can overfit because of the hard separator
- Nonlinear

-



Neural Networks

- Deep learning

Naïve bayes
Logistic regression
Decision tree
Support vector
Knn

Tf-idf in order to get rid of useless words

ML Libraries

Friday, April 22, 2022 8:30 PM

Related:Apache Spark

- Open source data processing engine for machine learning and AI applications
- Streaming data, graph data, machine learning and ai

Tensorflow

- C++ Cuda python
- High performance model
- complex

Keras

- Basically runs on top of tensorflow. Makes tensorflow easier to use
- Written in python

Parallel machine learning distributed

Thursday, April 21, 2022 9:50 PM

Elephas
- Keras extension
- Scale with Spark

FairScale
- MENTIONED BY INTERVIEWER
- Pytorch extension
- Parallelism
- Shard training
- Optimization at scale
- GPU memory optimization
- GPU speed optimization

TensorflowOnSpark
- Tensorflow deeplearning along with apache spark and apache hadoop
- Distributed deep learning on GPU and CPU
- Spark Clusters

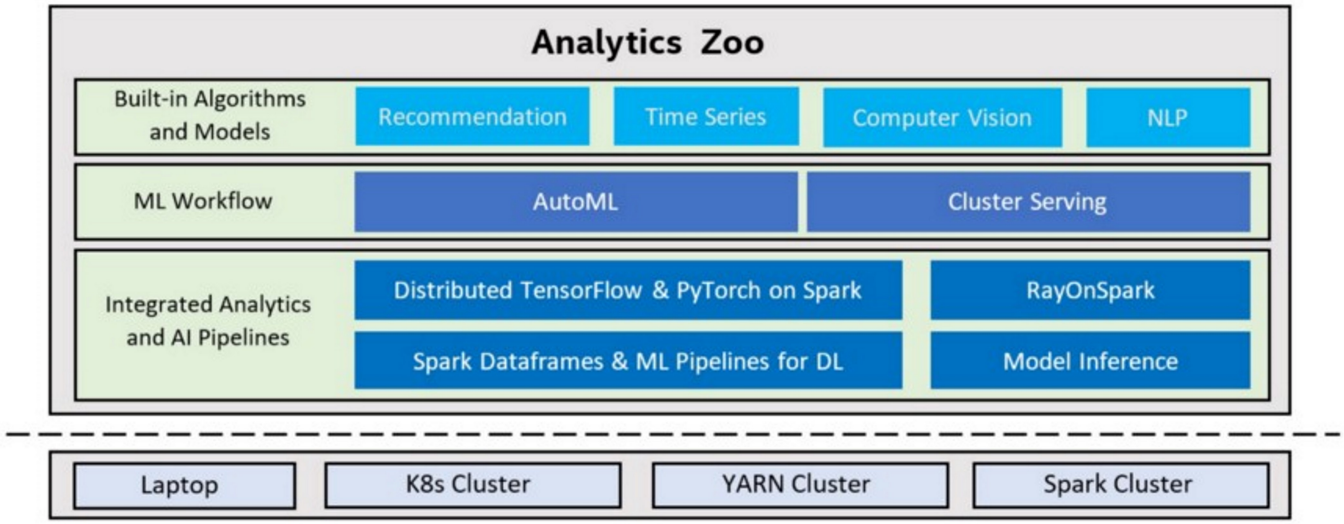
Deep Speed
- Gpu dependent super scalable model for deep learning
- From single computers to super computer centers

Horovod
- Deep learning for tensorflow, keras, pytorch, apache mxnet
- Horovod is easy and fast

Mesh Tensorflow
- Layer over tensorflow
- Formalize distribution strategies
- Split between hardware/processors

Bigdl
- Deep learning for apache spark
- Run directly on top of spark or hadoop clusters
- High performance for intel computing
- Modeled after torch
- Efficiently scale out

Analytics zoo



scales TensorFlow, Keras and PyTorch to distributed big data (using Spark, Flink & Ray).

-