**22/8/2023.**

Notes.

Only explanation for Curriculum, No theory today.

Tomorrow onwards lessons.

0%.

**23/8/2023.**

Types of Attacks.

Essential Terms.

Top 10 Notorious Hackers.

**Mitnick** **-** White hat

federal authorities were so concerned about his capabilities that when he was incarcerated again in 1995, Mitnick told CNN he was held in solitary confinement for a time out of concern that even proximity to a telephone could allow him to continue hacking.

Following his prison term, Mitnick became a white-hat hacker, using his expertise to legally help businesses track people trying to break into their systems. For the past decade, he was the chief hacking officer and partial owner of the tech security firm KnowBe4, founded by his close friend and business partner, Stu Sjouwerman.

**Anonymous -** Grey hat

Anonymous first became associated with hacking in 2008 following a series of actions against the Church of Scientology known as Project Chanology.

4chan users organized a raid against the Church in retaliation, prank-calling its hotline, sending [black faxes](https://en.wikipedia.org/wiki/Black_fax) designed to waste ink cartridges, and launching [DDoS](https://en.wikipedia.org/wiki/DDoS) attacks against its websites.

**Adrian Lamo -** Grey hat

Adrian Lamo was a controversial figure in the hacking world, known for challenging companies to re-think how secure they were.2 He was a hero in the hacker community for years and was known for exchanging messages with U.S. Army intelligence analyst Chelsea Manning.0

**Albert Gonzalez -** Black hat

**Albert Gonzalez** (born 1981) is an American [computer hacker](https://en.wikipedia.org/wiki/Hacker_(computer_security)), [computer criminal](https://en.wikipedia.org/wiki/Computer_criminal) and [police informer](https://en.wikipedia.org/wiki/Informant), who is accused of masterminding the combined [credit card theft](https://en.wikipedia.org/wiki/Carding_(fraud)) and subsequent reselling of more than 170 million card and [ATM](https://en.wikipedia.org/wiki/Automated_teller_machine) [numbers](https://en.wikipedia.org/wiki/Bank_card_number) from 2005 to 2007, the biggest such fraud in history. Gonzalez and his accomplices used [SQL injection](https://en.wikipedia.org/wiki/SQL_injection) to deploy [backdoors](https://en.wikipedia.org/wiki/Backdoor_(computing)) on several corporate systems in order to launch [packet sniffing](https://en.wikipedia.org/wiki/Packet_sniffing) (specifically, [ARP Spoofing](https://en.wikipedia.org/wiki/ARP_Spoofing)) attacks which allowed him to [steal computer data](https://en.wikipedia.org/wiki/Hacker_(computer_security)) from internal corporate networks.

**Matthew Bevan and Richard Pryce -** Grey hat

Matthew Bevan and Richard Pryce were a team of British hackers who hacked into multiple military networks in 1996, including Griffiss Air Force Base, the Defense Information System Agency, and the Korean Atomic Research Institute (KARI).

Bevan and Pryce demonstrated that even military networks are vulnerable.

**Jeanson James Ancheta -** Grey hat

Jeanson James Ancheta had no interest in hacking systems for credit card data or crashing networks to deliver social justice. Instead, Ancheta was curious about the use of bots—software-based robots that can infect and ultimately control computer systems. Using a series of large-scale "[botnets](https://www.kaspersky.com/resource-center/threats/botnet-attacks)," he was able to compromise more than 400,000 computers in 2005.

**Michael Calce -** Presently White hat

Michael Calce, also known as MafiaBoy, is a former computer hacker and security expert from Île Bizard, Quebec. In February 2000, he launched a series of highly publicized denial-of-service attacks against large commercial websites, including Yahoo!, Fifa.com, Amazon.com, Dell, Inc., E\*TRADE, eBay, and CNN

Calce now works as a security consultant for companies trying to protect their online systems.

**Kevin Poulsen -** Former Black hat

In 1983, a 17-year-old Poulsen, using the alias Dark Dante, hacked into ARPANET, the Pentagon’s computer network. Although he was quickly caught, the government decided not to prosecute Poulsen, who was a minor at the time. Instead, he was let off with a warning.Poulsen didn’t heed this warning and continued hacking. In 1988, Poulsen hacked a federal computer and dug into files pertaining to the deposed president of the Philippines, Ferdinand Marcos. When discovered by authorities, Poulsen went underground. While he was on the run, Poulsen kept busy, hacking government files and revealing secrets. Poulsen was arrested and barred from using a computer for three years.

He has since converted to white hat hacking and journalism, writing about cybersecurity and web-related socio-political causes for [Wired](https://www.wired.com/author/kevin-poulsen/), The Daily Beast and his own blog Threat Level. Paulson also teamed with other leading hackers to work on various projects dedicated to social justice and freedom of information.

**Jonathan James -** Grey hat

James was arrested in 2000 and was sentenced to a six months house arrest and banned from recreational computer use. However, a probation violation caused him to serve six months in jail. Jonathan James became the youngest person to be convicted of violating cyber crime laws. In 2007, TJX, a department store, was hacked and many customer’s private information were compromised. Despite a lack of evidence, authorities suspect that James may have been involved.

James committed suicide by gunshot. According to the [Daily Mail](https://www.dailymail.co.uk/news/article-2262831/Revealed-Aaron-Swartz-prosecutor-drove-hacker-suicide-2008-named-cyber-crime-case.html), his [suicide note](https://www.dailymail.co.uk/news/article-2262831/Revealed-Aaron-Swartz-prosecutor-drove-hacker-suicide-2008-named-cyber-crime-case.html) stated, “I have no faith in the 'justice' system. Perhaps my actions today, and this letter, will send a stronger message to the public. Either way, I have lost control over this situation, and this is my only way to regain control.”

**Astra -** Black hat

58-year-old Greek mathematician. Reportedly, he had been hacking into the Dassault Group, for almost half a decade. During that time, he stole cutting edge weapons technology software and data which he then sold to 250 individuals around the world. His hacking cost the Dassault Group $360 million in damages. No one knows why his complete identity has never been revealed.

**24/08/2023**

Vulnerable port 20,21.

| The FTP protocol employs a pair of connections between the client and server. Connections to the FTP server are initiated over the command and control channel which is established from an arbitrary port on the FTP client machine to (usually) port 21 on the FTP server machine.  Next, a second "data channel" connection is created. This is either made from the FTP server's port 20 to a client-specified port on the client machine (in the case of Active FTP), or from an arbitrary port on the client machine to port 20 on the FTP server machine.  In other words, the first "command and control" connection of an FTP session is made from the client to the server (to port 21 by default) then the second connection, as specified by the commands carried over the first connection, is made either the client to the server, or back from the server to the client. |
| --- |

* Brute-forcing passwords
* Anonymous authentication (it’s possible to log into the FTP port with “anonymous” as the username and password)
* Cross-site scripting
* Directory traversal attacks

Servers with port 22 open are prone to brute-force attacks. The best option is to change the SSH port to a higher available port (31000, 41762, etc). If connection is always going to be from the same IPs, allowing only connection from those IPs is advisable.

Port 23

Telnet is one of the oldest Internet protocols and the most popular program for remote access to Unix machines. It has numerous security vulnerabilities

Buffer overflow

Backdoors

Port 25

SMTP

Open relay

Email Spoofing

Port 53

DNS

Cache Poisioning

DDos Amplification

Port 69

TFTP lacks authentication

Data Exposure

Port 80  
XSS Cross site scripting

SQL injection

Port 110

Email Deletion

Password Exposure

Port 123

Reflection Attacks

Time Spoofing

Port 143 IMAP

Email Hijacking

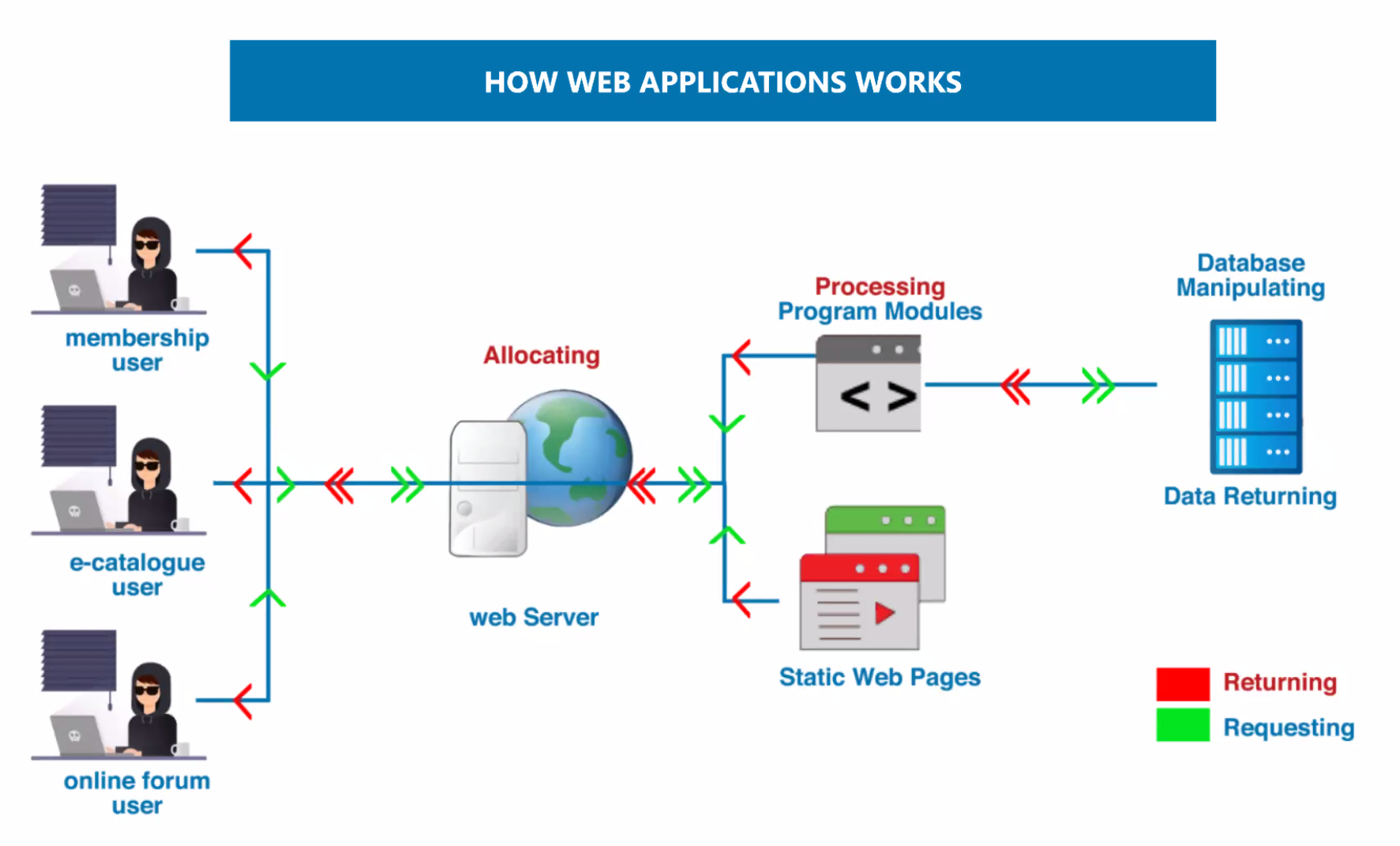
Credential Theft

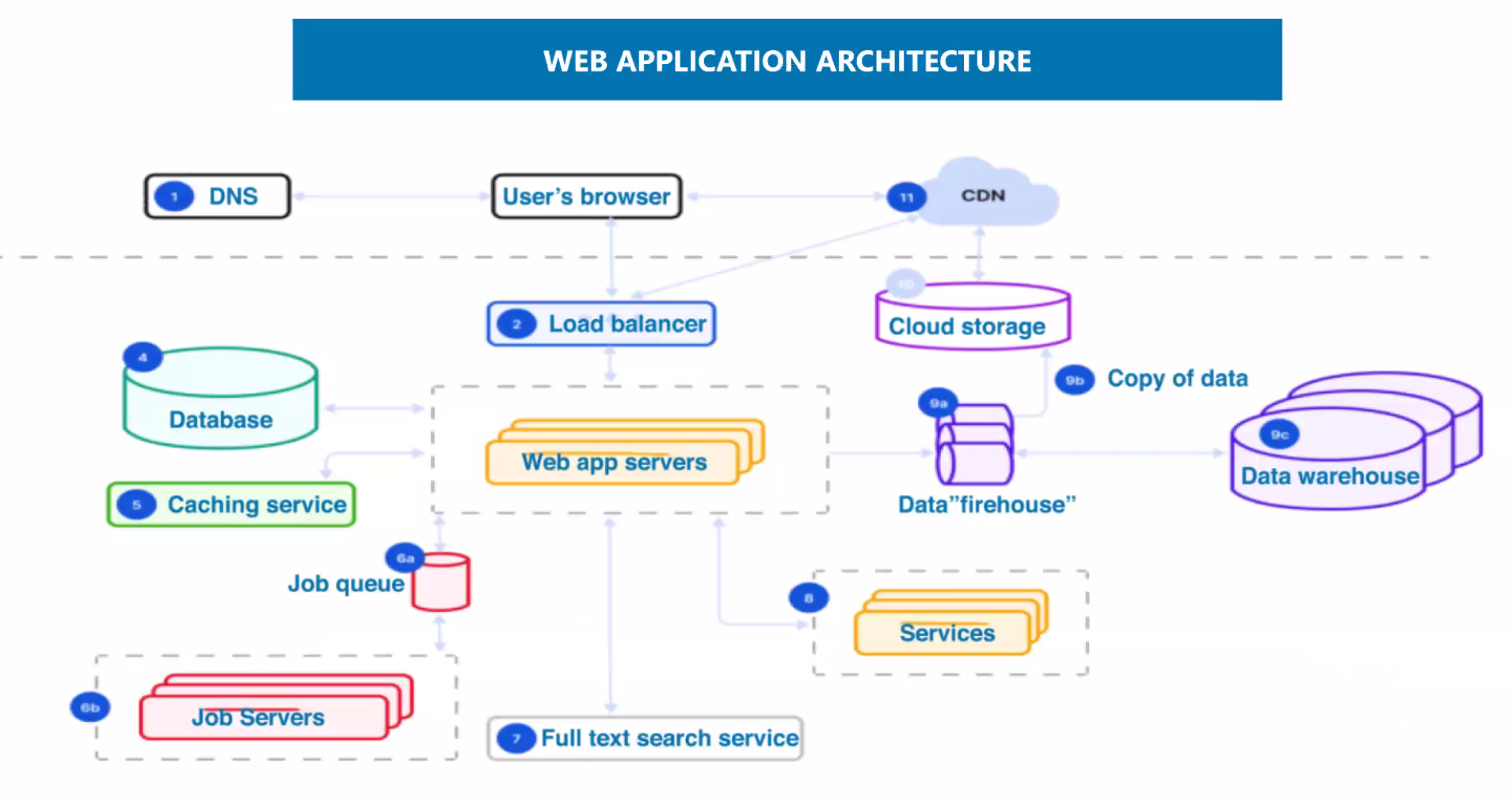
**25/08/2023.**

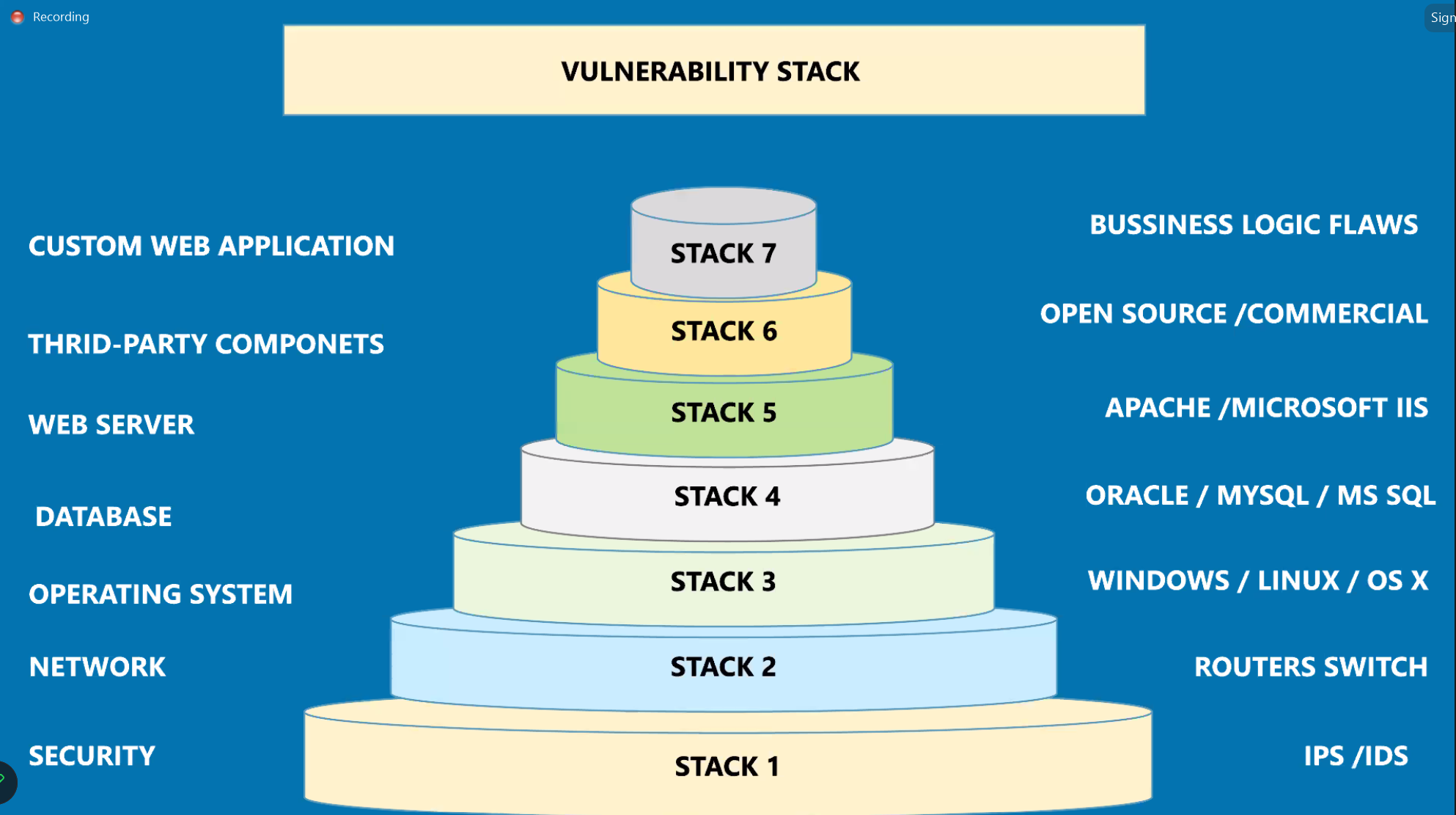
Web Application Hacking.

Accessed over network. (Static and Dynamic Types.)

Both type of Webapp has different vulnerabilities.







What is the OWASP Top 10?

The OWASP Top 10 is a regularly-updated report outlining security concerns for web application security, focusing on the 10 most critical risks. The report is put together by a team of security experts from all over the world. OWASP refers to the Top 10 as an ‘awareness document’ and they recommend that all companies incorporate the report into their processes in order to minimize and/or mitigate security risks.

Below are the security risks reported in the OWASP Top 10 2017 report:

1. Injection

Injection attacks happen when untrusted data is sent to a code interpreter through a form input or some other data submission to a web application. For example, an attacker could enter SQL database code into a form that expects a plaintext username. If that form input is not properly secured, this would result in that SQL code being executed. This is known as an [SQL injection attack](https://www.cloudflare.com/learning/security/threats/sql-injection/)

2. Broken Authentication

Vulnerabilities in authentication (login) systems can give attackers access to user accounts and even the ability to compromise an entire system using an admin account. For example, an attacker can take a list containing thousands of known username/password combinations obtained during a [data breach](https://www.cloudflare.com/learning/security/what-is-a-data-breach/) and use a script to try all those combinations on a login system to see if there are any that work.

Some strategies to mitigate authentication vulnerabilities are requiring [two-factor authentication (2FA)](https://www.cloudflare.com/learning/access-management/what-is-two-factor-authentication/) as well as limiting or delaying repeated login attempts using [rate limiting](https://www.cloudflare.com/rate-limiting/)

3. Sensitive Data Exposure

If web applications don’t protect sensitive data such as financial information and passwords, attackers can gain access to that data and sellor utilize it for nefarious purposes. One popular method for stealing sensitive information is using an [on-path attack](https://www.cloudflare.com/learning/security/threats/on-path-attack/).

Data exposure risk can be minimized by [encrypting](https://www.cloudflare.com/learning/ssl/what-is-encryption/) all sensitive data as well as disabling the [caching](https://www.cloudflare.com/learning/cdn/what-is-caching/)\* of any sensitive information. Additionally, web application developers should take care to ensure that they are not unnecessarily storing any sensitive data.

\*Caching is the practice of temporarily storing data for re-use. For example, web browsers will often cache webpages so that if a user revisits thosepages within a fixed time span, the browser does not have to fetch the pages from the web.

4. XML External Entities (XEE)

This is an attack against a web application that parses XML\* input. This input can reference an external entity, attempting to exploit a vulnerability in the parser. An ‘external entity’ in this context refers to a storage unit, such as a hard drive. An XML parser can be duped into sending data to an unauthorized external entity, which can pass sensitive data directly to an attacker.

The best ways to prevent XEE attacks are to have web applications accept a less complex type of data, such as JSON\*\*, or at the very least to patch XML parsers and disable the use of external entities in an XML application.

\*XML or Extensible Markup Language is a markup language intended to be both human-readable and machine-readable. Due to its complexity and security vulnerabilities, it is now being phased out of use in many web applications.

\*\*JavaScript Object Notation (JSON) is a type of simple, human-readable notation often used to transmit data over the internet. Although it was originally created for JavaScript, JSON is language-agnostic and can be interpreted by many different programming languages.

5. Broken Access Control

[Access control](https://www.cloudflare.com/learning/access-management/what-is-access-control/) refers a system that controls access to information or functionality. Broken access controls allow attackers to bypass authorization and perform tasks as though they were privileged users such as administrators. For example a web application could allow a user to change which account they are logged in as simply by changing part of a url, without any other verification.

Access controls can be secured by ensuring that a web application uses authorization tokens\* and sets tight controls on them.

\*Many services issue authorization tokens when users log in. Every privileged request that a user makes will require that the authorization token be present. This is a secure way to ensure that the user is who they say they are, without having to constantly enter their login credentials.

6. Security Misconfiguration

Security misconfiguration is the most common vulnerability on the list, and is often the result of using default configurations or displaying excessively verbose errors. For instance, an application could show a user overly-descriptive errors which may reveal vulnerabilities in the application. This can be mitigated by removing any unused features in the code and ensuring that error messages are more general.

7. Cross-Site Scripting

[Cross-site scripting](https://www.cloudflare.com/learning/security/threats/cross-site-scripting/) vulnerabilities occur when web applications allow users to add custom code into a url path or onto a website that will be seen by other users. This vulnerability can be exploited to run malicious JavaScript code on a victim’s browser. For example, an attacker could send an email to a victim that appears to be from a trusted bank, with a link to that bank’s website. This link could have some malicious JavaScript code tagged onto the end of the url. If the bank’s site is not properly protected against cross-site scripting, then that malicious code will be run in the victim’s web browser when they click on the link.

8. Insecure Deserialization

This threat targets the many web applications which frequently serialize and deserialize data. Serialization means taking objects from the application code and converting them into a format that can be used for another purpose, such as storing the data to disk or streaming it. Deserialization is just the opposite: converting serialized data back into objects the application can use. Serialization is sort of like packing furniture away into boxes before a move, and deserialization is like unpacking the boxes and assembling the furniture after the move. An insecure deserialization attack is like having the movers tamper with the contents of the boxes before they are unpacked.

An insecure deserialization exploit is the result of deserializing data from untrusted sources, and can result in serious consequences like [DDoS attacks](https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/) and remote code execution attacks. While steps can be taken to try and catch attackers, such as monitoring deserialization and implementing type checks, the only sure way to protect against insecure deserialization attacks is to prohibit the deserialization of data from untrusted sources.

9. Using Components With Known Vulnerabilities

Many modern web developers use components such as libraries and frameworks in their web applications. These components are pieces of software that help developers avoid redundant work and provide needed functionality; common example include front-end frameworks like React and smaller libraries that used to add share icons or a/b testing. Some attackers look for vulnerabilities in these components which they can then use to orchestrate attacks. Some of the more popular components are used on hundreds of thousands of websites; an attacker finding a security hole in one of these components could leave hundreds of thousands of sites vulnerable to exploit.

10. Insufficient Logging And Monitoring

Many web applications are not taking enough steps to detect data breaches. The average discovery time for a breach is around 200 days after it has happened. This gives attackers a lot of time to cause damage before there is any response. OWASP recommends that web developers should implement logging and monitoring as well as incident response plans to ensure that they are made aware of attacks on their applications.

TASK - 3

* [**A01:2021-Broken Access Control**](https://owasp.org/Top10/A01_2021-Broken_Access_Control/) moves up from the fifth position; 94% of applications were tested for some form of broken access control. The 34 Common Weakness Enumerations (CWEs) mapped to Broken Access Control had more occurrences in applications than any other category.

CWE 285 - The product does not perform or incorrectly performs an authorization check when an actor attempts to access a resource or perform an action.

Business Impact: It can lead to malicious actors getting in and wreak havoc on the system in place. They can steal data, pretend to be someone else, escalate privileges etc.

* [**A02:2021-Cryptographic Failures**](https://owasp.org/Top10/A02_2021-Cryptographic_Failures/) Weaknesses in this category are related to the design and implementation of data confidentiality and integrity. Frequently these deal with the use of encoding techniques, encryption libraries, and hashing algorithms. The weaknesses in this category could lead to a degradation of the quality data if they are not addressed.

Business Impact: if a flaw is discovered with hardware-implemented cryptography, the flaw cannot be fixed in most cases without a recall of the product, because hardware is not easily replaceable like software. Second, because the hardware product is expected to work for years, the adversary's computing power will only increase over time.

* [**A03:2021-Injection**](https://owasp.org/Top10/A03_2021-Injection/) slides down to the third position. 94% of the applications were tested for some form of injection, and the 33 CWEs mapped into this category have the second most occurrences in applications. Cross-site Scripting is now part of this category in this edition.

CWE 20 - improper input Validation.

Buisness impact:

Sensitive information can be stolen,Attackers can execute arbitrary commands,Because of this, financial loss, Operational Block and Reputation Damage will follow.

* [**A04:2021-Insecure Design**](https://owasp.org/Top10/A04_2021-Insecure_Design/) is a new category for 2021, with a focus on risks related to design flaws. If we genuinely want to “move left” as an industry, it calls for more use of threat modeling, secure design patterns and principles, and reference architectures.

CWE-235: Improper Handling of Extra Parameters

The product does not handle or incorrectly handles when the number of parameters, fields, or arguments with the same name exceeds the expected amount.

Business Impact: Data Exposure,Exploitation of Weaknesses,Reputation Damage.

* [**A05:2021-Security Misconfiguration**](https://owasp.org/Top10/A05_2021-Security_Misconfiguration/) moves up from #6 in the previous edition; 90% of applications were tested for some form of misconfiguration. With more shifts into highly configurable software, it’s not surprising to see this category move up. The former category for XML External Entities (XXE) is now part of this category.

Debugging messages help attackers learn about the system and plan a form of attack.

ASP .NET applications can be configured to produce debug binaries. These binaries give detailed debugging messages and should not be used in production environments. Debug binaries are meant to be used in a development or testing environment and can pose a security risk if they are deployed to production.

CWE-260: Password in Configuration File

The product stores a password in a configuration file that might be accessible to actors who do not know the password.

Business impact: An attacker could gain access to this file and learn the stored password or worse yet, change the password to one of their choosing.

Unauthorized Access,Data Breach,Credential Exposure,Compliance Violations.

**28/8/2023.**

**10 Vulnerabilities that are not included in OWASP top 10**

**Race Conditions**: These occur when the behavior of a software system depends on the sequence or timing of events, which can lead to unauthorized access or data corruption.

**Content Spoofing**: Attackers manipulate content to deceive users, often leading to phishing attacks or delivering malicious content.

**Cross-Protocol Attacks**: Exploiting interactions between different protocols to compromise security.

**Credential Stuffing:** Attackers use previously leaked credentials to gain unauthorized access to other accounts due to users reusing passwords.

**Business Logic Vulnerabilities**: These arise from flawed business rules in applications, leading to unauthorized actions or data manipulation.

**DOM-based Attacks**: Attackers exploit vulnerabilities in the Document Object Model (DOM) of a web page to execute malicious actions in the user's browser.

**No Memory Limit:** Some Applications can go on using more and more ram without freeing up unused memory. This can go on till the computer crashes.

**Insecure Transfer**: Data Transfer could be vulnerable to attacks from outside if using outdated software.

**Malicious Insider Threats:** Insiders with authorized access can intentionally misuse privileges to steal data or cause harm.

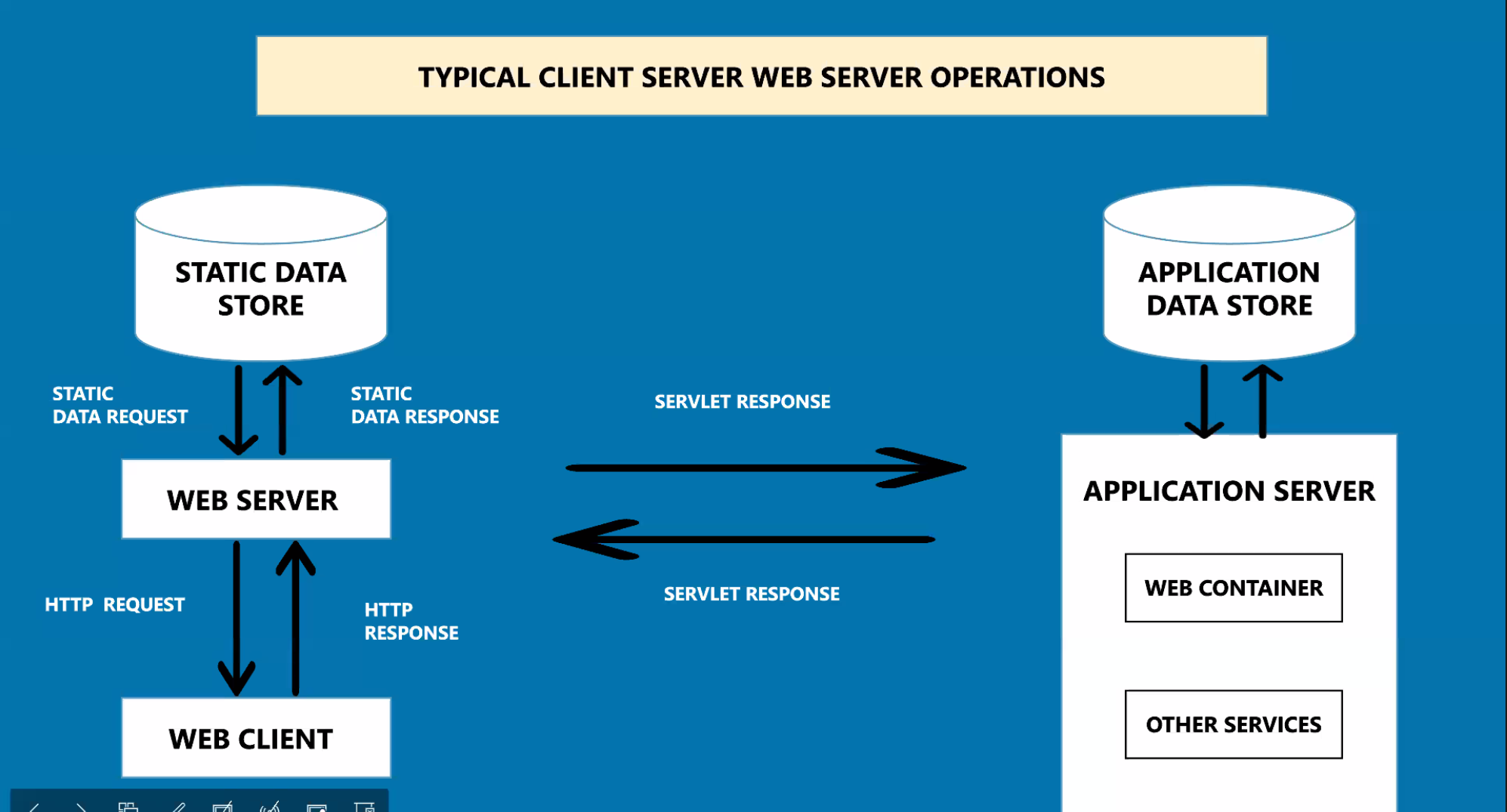
**API Abuse:** Malicious use of APIs to perform unauthorized actions, often exploiting poorly implemented or documented endpoints.

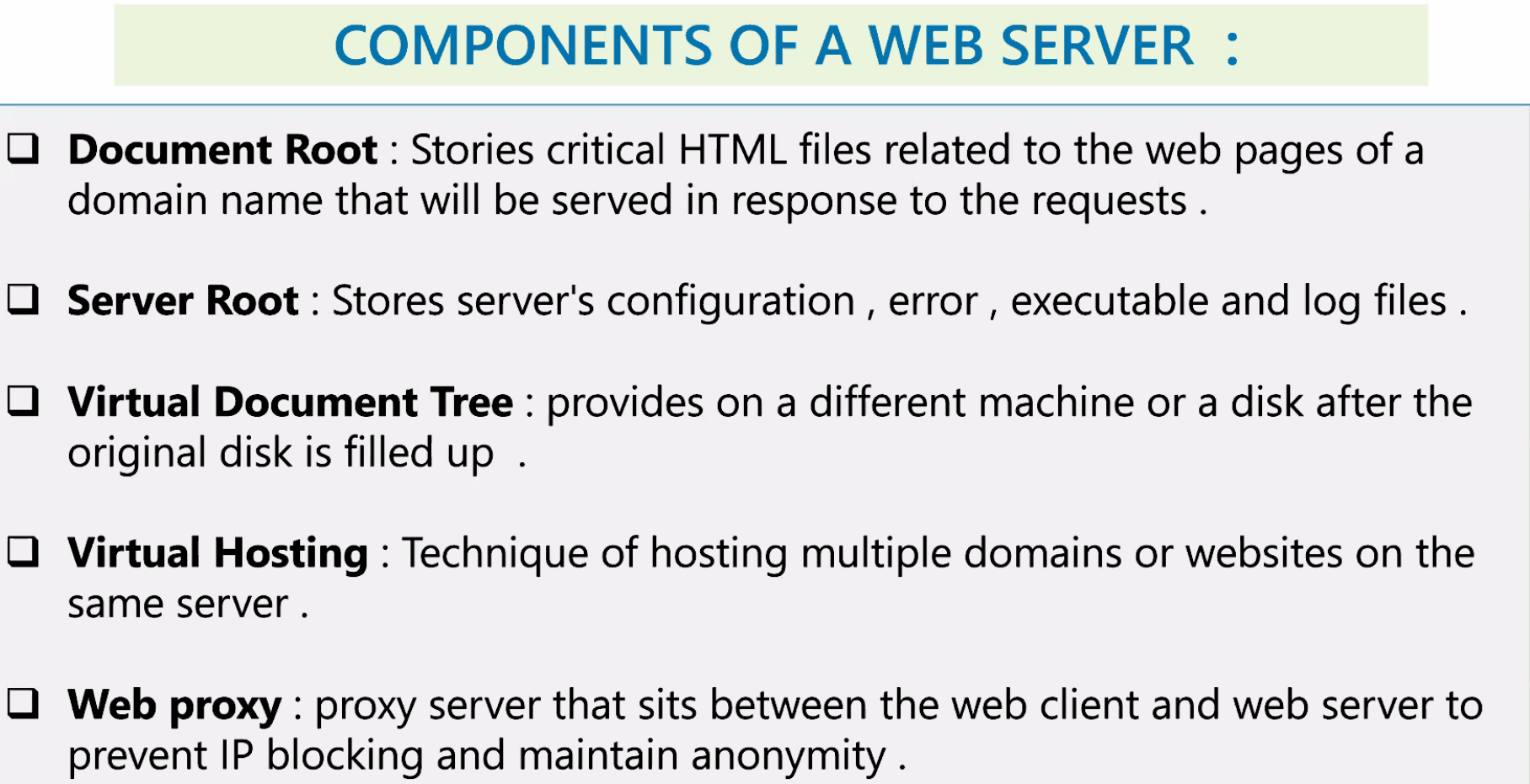
**29/8/2023**

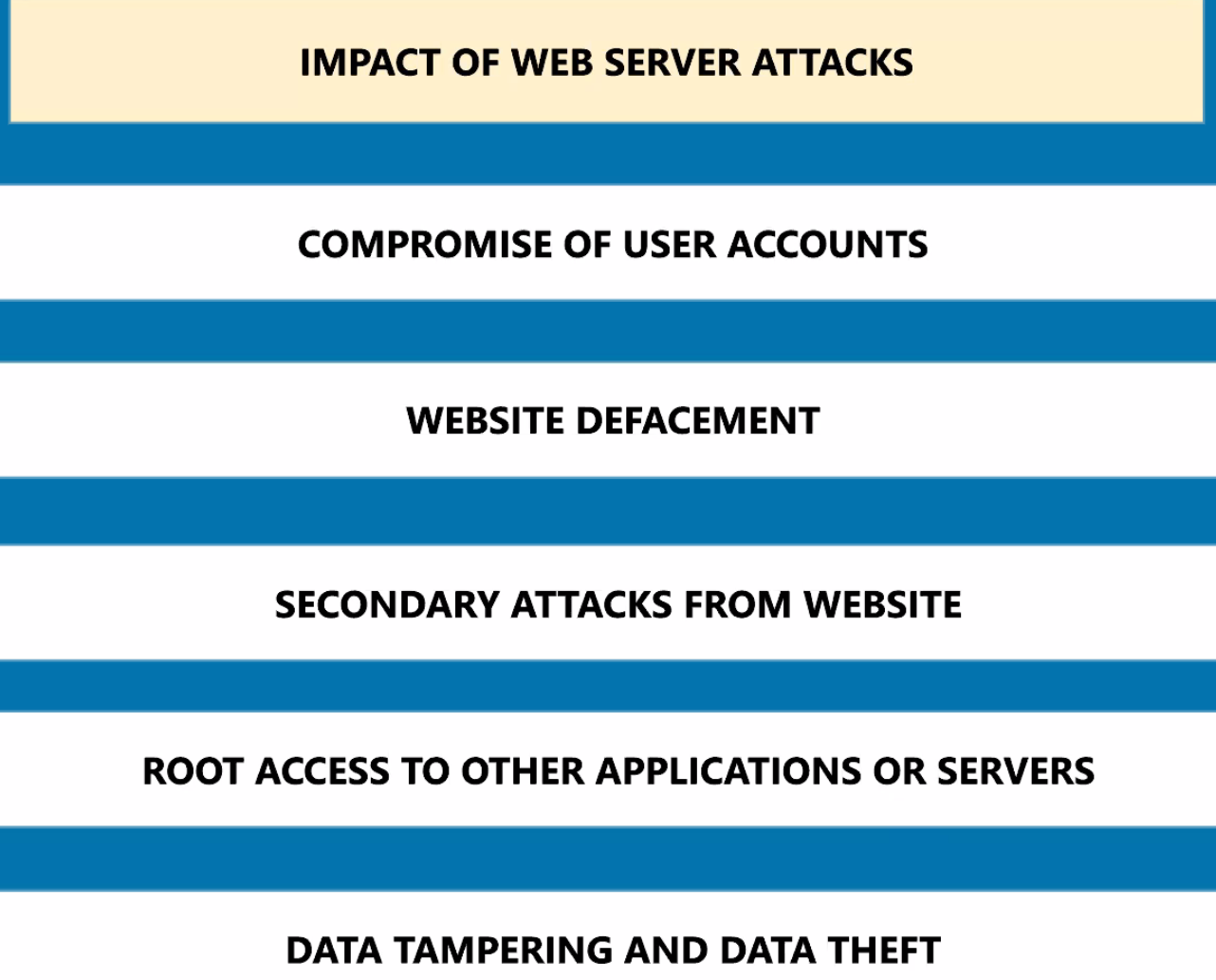
WebServer Concepts.

Webserver Attacks and countermeasures.

Attack Tools.







**Task 6**

### 

### **1.URL INTERPRETATION ATTACK**

This attack is also called URL poisoning as the attackers manipulates the URL by changing its semantics but keeping the syntax intact. The parameters of the URL are adjusted so that information beyond what is intended can be retrieved from the web server.

To understand how this type of attack is perpetrated, we can take an example of an email. Using URL interpretation, the URL that carries the request to fetch the details of a user and send it to the alternative email id provided, can be modified to fetch the detail of another user. Thus, URL interpretation attack makes the information of other users vulnerable.

**2.SQL INJECTION ATTACK**

. An SQL query with parameters from the URL is fed to the database that has the ability to alter the data. The stored procedures in the database can also be executed through SQL injection and database can be made to do things, it is intended to do only when desired by the authorized personnel.

The vulnerability that this type of attack exploits is the scenario where the SQL query is permitted to be executed without validating the input data. Websites that are most likely to be attacked using this type of attack are e-commerce websites that have huge database comprising users™ information.

### **3.INPUT VALIDATION ATTACK**

Input validation attack is an attack on the web server where the server executes a code injected by a hacker to the web server or the database server. There are many input types that need to be validated before execution including data type, data ranges, and others. By executing the code with inputs that are not validated, information can be retrieved or modified by the attacker.

This attack is done by bypassing the client side checking using JavaScript code. Tampering of the hidden file is also possible with this attack. Data to the web server may be sent in a variety of ways including the URL, HTTP headers, POST requests, and cookies. Any negligence while writing the code and trusting the data received from the users can lead to such attacks.

### **4.BUFFER OVERFLOW ATTACKS**

Buffer Overflow attack implies the deliberate overflowing of the buffer memory that is reserved for the users™ input. When an application awaits a users™ input, it allocates a stack with a memory location where the input data by the user is fed. The attackers flood this space by writing arbitrary data so that the memory stack is full and the users deny the service. This is one of the ways to perform denial of service attack which is dealt with in more detail further.

Another aspect of this type of attack is that the hacker can feed an executable command in the stack. Although, the surety of the execution of the command is dependent on the return address that is specified by the hacker. After the stack recovers from the crash, it goes to the return address and if it has been changed and replaced with one that falls within the desired range, the command may execute and grant entry to certain sections of the web server.

### **5.IMPERSONATION ATTACKS**

Impersonation attack is also called IP spoofing where the hacker pretends to be accessing the web server with an IP that is actually impersonating an IP that has the access to the web server. There are special programs that the hackers make use of, to create an IP packet that appears to be originating from the intranet and hence gain entry to the section of the web server that is intended to be accessed only by the authorized personnel.

Impersonation attacks exploit the vulnerability of the authentication protocols and get unauthorized access to the web servers and the databases. Such attacks can be countered by having a strong authentication module and identification of the traffic coming towards the server.

### **6.PASSWORD-BASED ATTACKS**

The authentication system of a web server is often based on the password that identifies a valid user and grants access to the web server. If the hacker can, by any means, get your username and password, he or she can access the information that only you are supposed to access. The older applications do not have strong authentication system and this makes it easy for the eavesdroppers to get through the authentication process.

Breaking a password is not easy and there are certain algorithms that the hackers deploy to guess the password and gain access to the network. Once hackers get access to the network, they can modify the configuration to make it easy for them to hack the network again if normalcy is restored in the website by attempts from the network engineers.

### **7.DENIAL OF SERVICE ATTACKS**

Denial of service attack (DOS) is an attack where the server denies serving the users with a response to their request. This attack is performed by several means and buffer flow is one of them. It is an effective and naturally one of the most popular ways of attacking the web server. The attackers after gaining access to the network randomize the attention of the security system experts so that they do not become aware of the attack immediately so that they can exploit the web server in other ways.

DOS attacks are performed by overwhelming the web server in numerous ways including sending invalid data as input that causes application termination, flooding the web server with automated request causing a crash, blocking the traffic resulting in loss of access to the users. DOS attacks are categorized under volume attack, protocol attack, and Application layer attack.

### **8.BRUTE FORCE**

Brute Force, as the name suggests, implies cracking the username, password combination by using all possible iterations. This is a basic form of web server attack and is implemented when the hackers have a clue that weak passwords have been used in the authentication. The chance of brute force working is maximum when no other security measures are there besides password authentication.

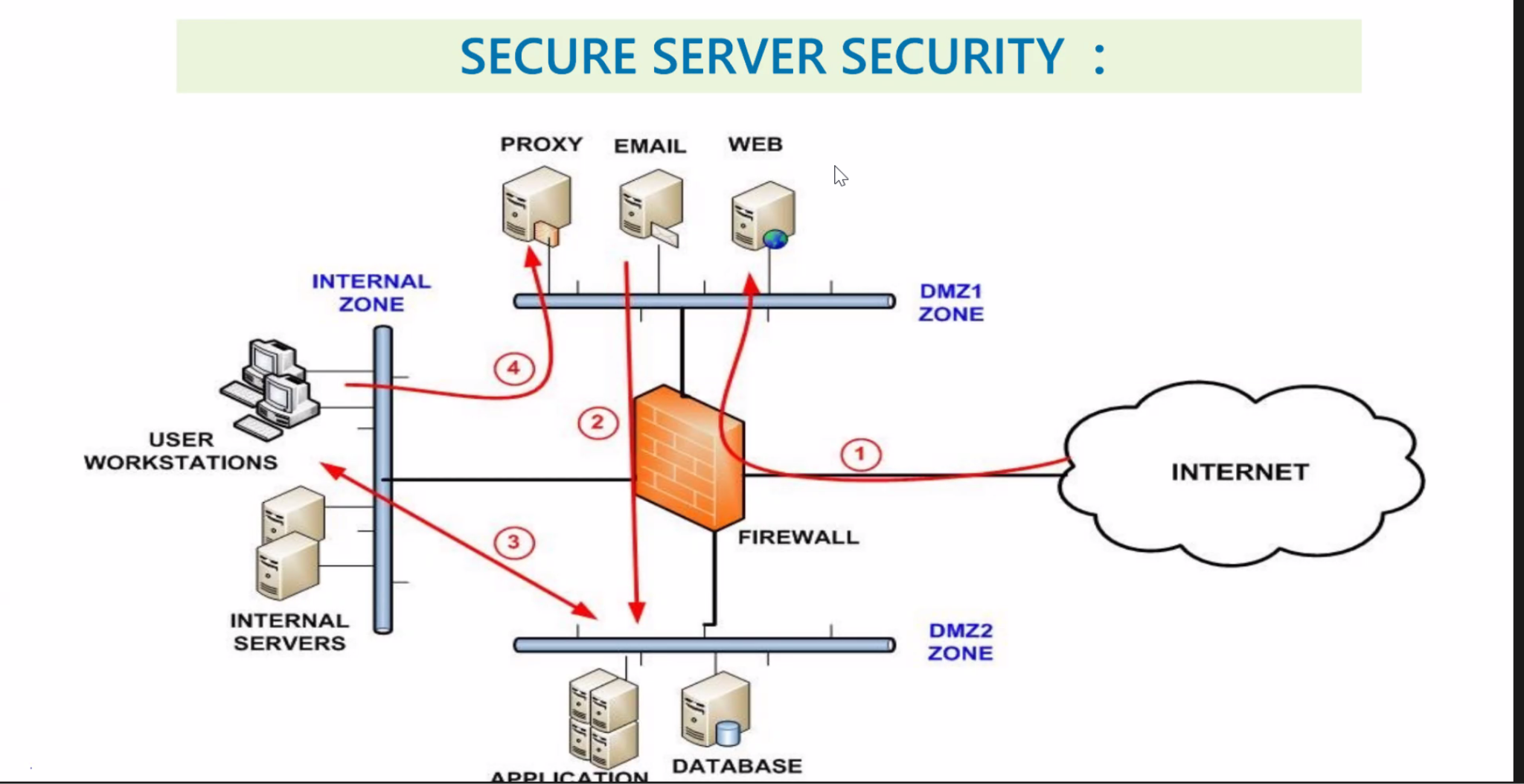
### **9.SOURCE CODE DISCLOSURE**

Through Source code disclosure attack, the attackers are able to retrieve the application files without using any parsing. The source code of the application is recovered and then it is analyzed to find loopholes that can be used to attack the web servers. It is often caused when the application is designed poorly or there are errors in the configuration.

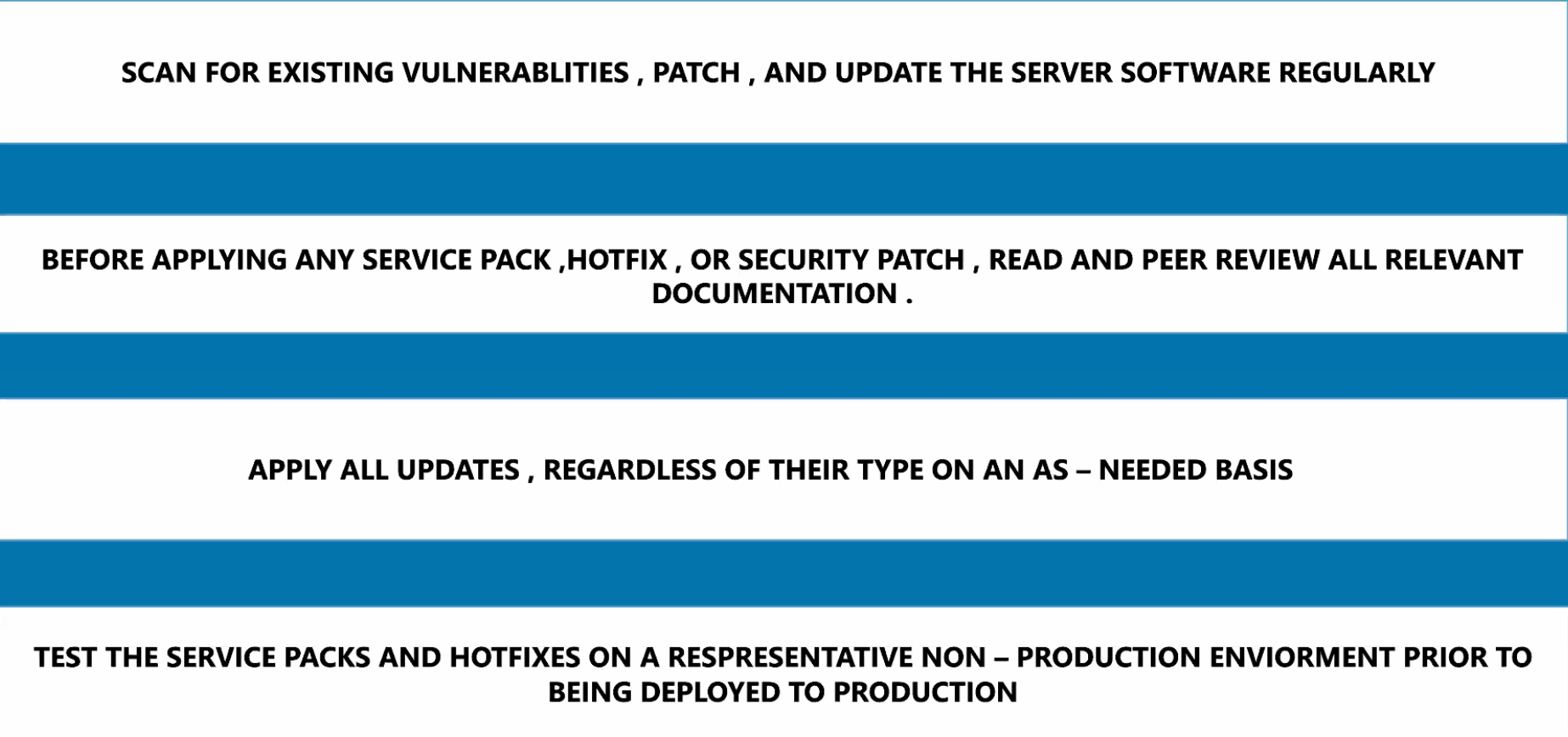
### **10. SESSION HIJACKING**

HTTP is a stateless protocol while all the web applications have states. When the tracking of these states is based on poor mechanism, session hijacking becomes easy for the hackers. It is also called cookie hijacking because a web server determines the session with a user based on the cookie. The cookie stored on the users™ computer is stolen by the hijacker by either intercepting it through the access to the network or through a previously saved cookie. Sniffing programs are used to perform this attack in an automated manner.

**30/08/2023**

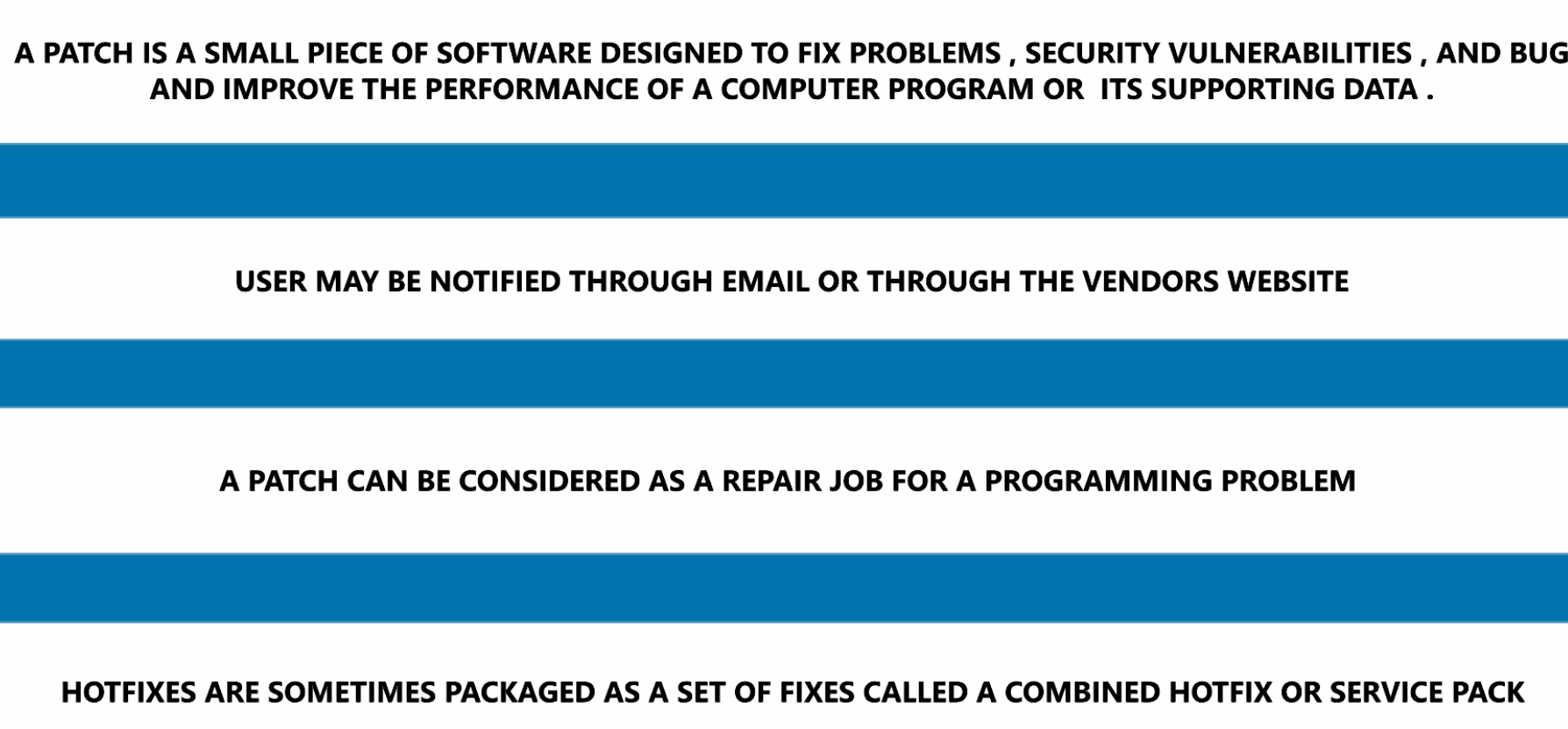
****

Preparing Countermeasures





Patch and hotfixes.



**Task 7**

**1. Inventory and Control of Hardware.**

We have to keep monitoring the Devices that are connected to the corporate assets like networks, email services,cloud applications, because any unprotected system is a potential exploit for hackers, who by way of scanners keep a tight watch on the organization.

**2. Inventory and Control of Software**

The Organization should manage all software on their networks in a way that only allows the installation of authorized software. For example: application whitelisting, Frequent Updation and Testing. Monitoring installed software regularly etc.

This is essential to prevent unwanted remote access.

**3. Secure Configurations for Hardware and Software**

security configurations of everyday devices like servers, laptops, and workstations should be established and taken care of regularly.

The members of the defending party should follow strict configuration management and implement change control processes to stop attackers from exploiting vulnerable services and settings.

**4. Continuous Vulnerability Assessment and Remediation**

Organisations should continuously assess and act on newly acquired information concerning their systems and the current landscape. For Example: software updates, patches, threat discovery, and security reports.

This information will help us identify and remove vulnerabilities in the networks we use regularly .

**5. Controlled Use of Administrative Privileges**

The companies are encouraged to use automated tools to monitor user behavior and track how administrator privileges are assigned and used to prevent unauthorized access to critical systems.

**6. Maintenance, Monitoring, and Analysis of Audit Logs**

Organizations are responsible for collecting, managing, and analyzing event logs. Doing this will allow them to detect any suspicious activities and investigate security incidents.

This has the benefit of not letting attackers hide their location and activities in the network even when attacking multiple times.

**7. Email and Web Browser Protections**

Organizations should use only supported web browsers and email clients to minimize attack vectors.

Email clients, for example, are a common entry point for numerous spoofing, phishing attacks. etc.

**8. Malware Defenses**

It is suggested that organisations used standard automated tools to monitor workstations, mobile devices and servers with antivirus, anti-spyware, and firewalls. etc.

**9. Limitation and Control of Network Ports, Protocols, and Services**

Tracking in ports, protocols, and services or network devices is another step taken by an organization to reduce the level of vulnerability available to attackers.

**10. Data Recovery Capability**

Companies should back up critical systems and data every week (at the minimum).

This is because attackers can often change the configuration, software, and data during a cyberattack. Without a reliable backup, we may not recover post-attack.

**11. Secure Configurations for Network Devices**

Organizations must create, implement, and manage security configurations of network devices such as routers, firewalls, and switches.

This is mainly because in general network infrastructure configurations are made with ease of deployment in mind and not security.

Network security devices are often left behind over time, allowing malicious hackers to exploit the flaws and get access to networks or exploit compromised machines to pose as trusted systems.

**12. Boundary Defense**

Businesses have to detect and correct information flow between networks of different trust levels while focusing on data that could damage security. We need to use technologies to control data flow and provide visibility across the environment.

**13. Data Protection**

Organizations are required to use appropriate processes and tools to eliminate data exfiltration risks while ensuring the integrity of sensitive information. This control outlines a combination of best practices concerning encryption, integrity protection, and data loss prevention techniques.

**14. Controlled Access Based on the Need to Know**

Tracking, controlling, and accessing critical assets is crucial—and it allows leaders to easily determine which people, computers, or applications should have the right to access them and at what level.

**15. Wireless Access Control**

Organizations need to implement certain tools and processes to effectively track and control how their employees use wireless local area networks (LANs), wireless client systems, and access points.

**16. Account Monitoring and Control**

The organisations must be responsible for managing the lifecycle of user accounts (creation, use, and deletion) to prevent creating opportunities for hackers to misuse them, accounts should be regularly reviewed, while accounts of former contractors and employees should be disabled as soon as they leave the company.

**17. Security Skills Assessment and Appropriate Training** It’s important to identify the specific knowledge and skills organizations need to strengthen security. We need to develop and execute plans to identify and fix gaps through organisational policy, planning, and training.

**18. Application Software Security**

Managing the security lifecycle of all software is another necessary activity.

This is important to detect and correct security weaknesses, it is important to make use of current versions of an app and that all relevant patches are installed.

**19. Incident Response and Management**

Organizations can enhance incident response by developing strategies, defining roles, training, and implementing management supervision to improve detection and damage control of assaults.

**20. Penetration Tests and Red Team Exercises**

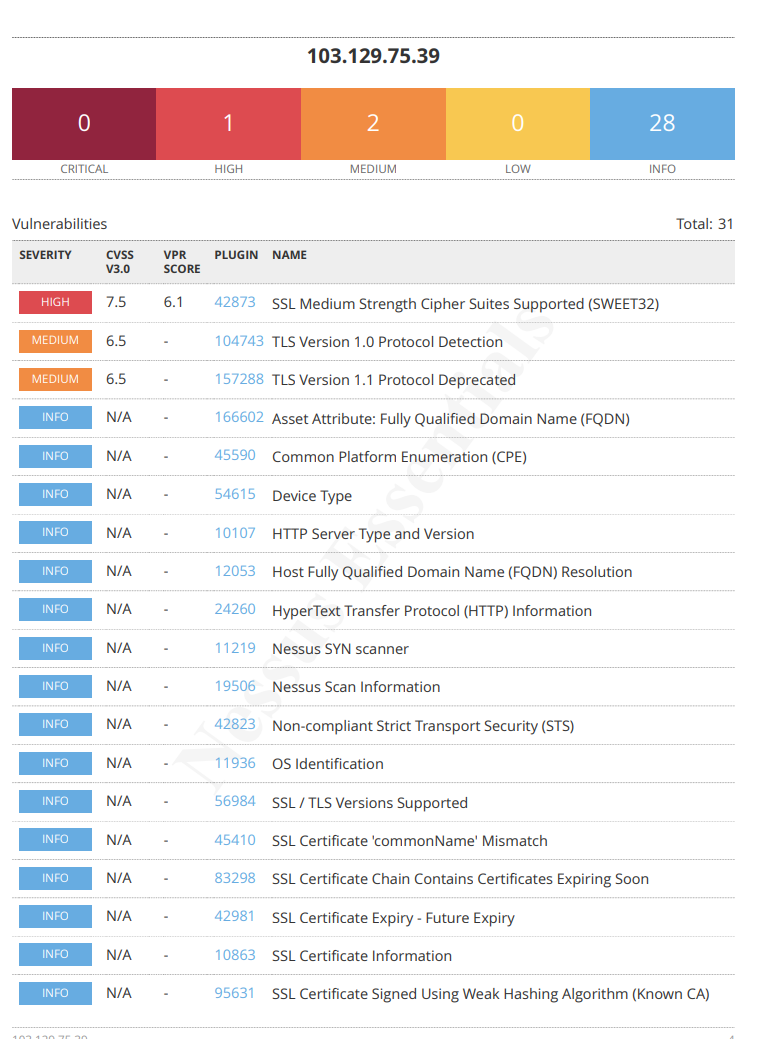
Organizations must assess defense security, including technology, processes, and expertise, through regular penetration tests to identify vulnerabilities and attack vectors for potential system exploitation.

**Task 8**

Scan any website to check the vulnerabilities in the website using

nessus and make a report on it:

1st host.

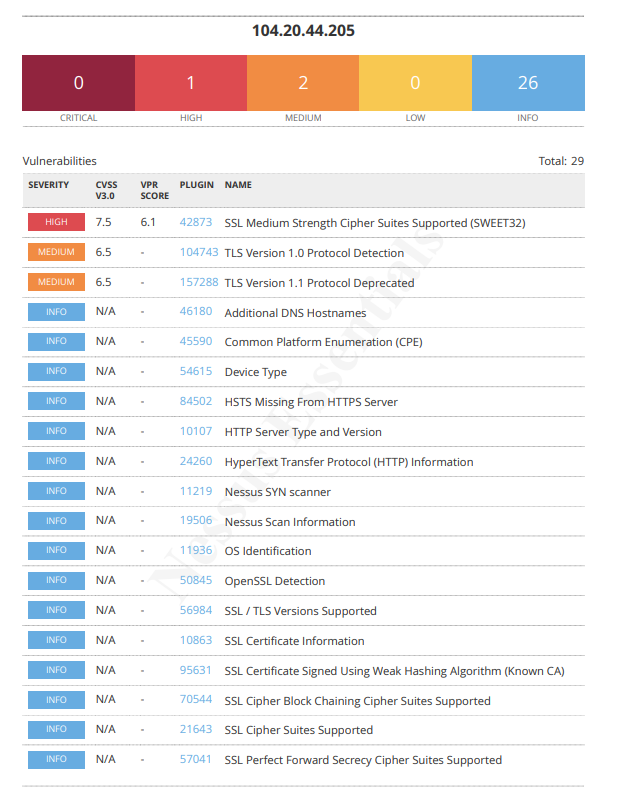


Non-info Vulnerabilities found:

**CVE-2016-2183 (medium risk) - medium strength cipher.**

**CWE-327. (Low risk) - deprecated TLS version. X2 (v1.0 and v1.1)**

2nd host.

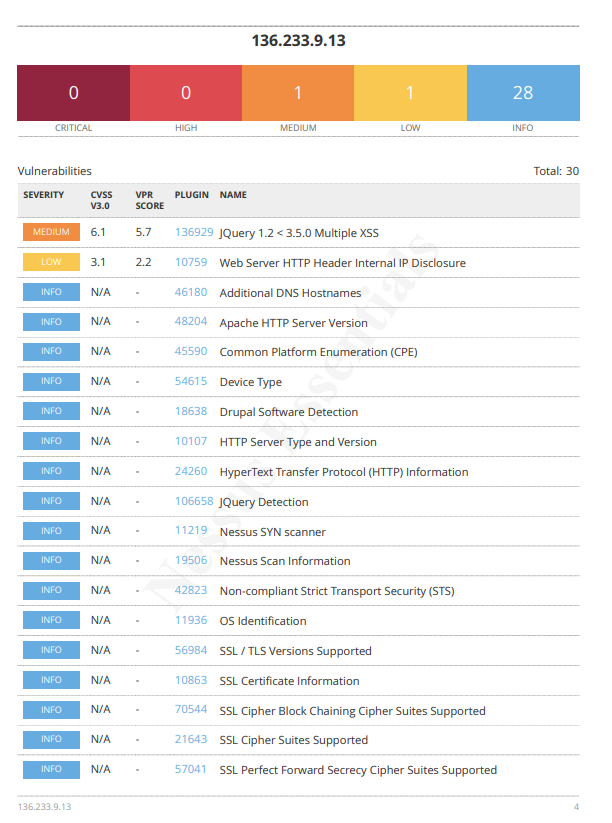


Vulnerability:

**CVE-2016-2183 - (High risk) SWEET32.**

**CWE-327. (Low risk) - deprecated TLS version.**

Host 3

****

Vulnerability:

## **CVE-2020-11022 - (medium risk) - Multiple XSS**

**CWE-200 - (low risk) - missing HTTP header.**