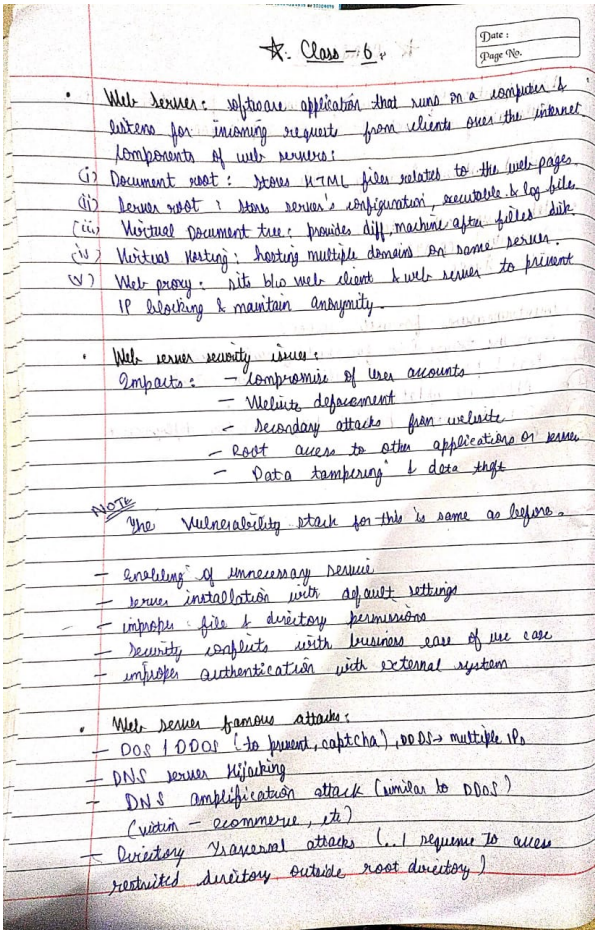
Class 6 - 29/08/2023

Today we studied about the web server and its components, web server security issues, web servers famous attacks, web defacement, web server misconfiguration, various web servers and application attacks and in the end we studied about the web server attack methodology.

Here are the screenshots from the short notes i made from the class:



☆ Class - 6

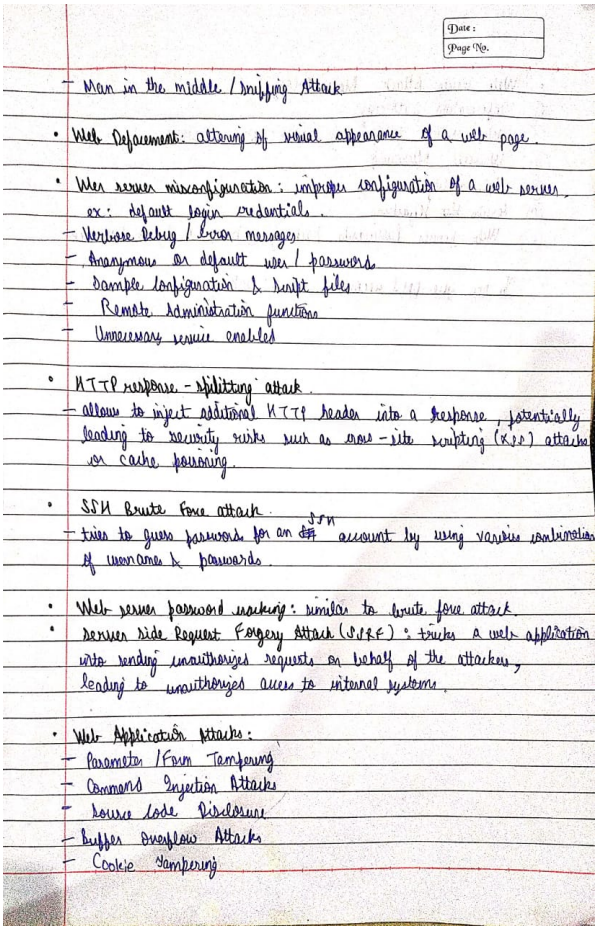
- Web Servers: software application that runs on a computer & listens for incoming requests from clients over the internet
Components of web servers:
(i) Document root: stores HTML files related to the web pages.
(ii) Server root: stores server's configuration, executable & log files
(iii) Virtual Document tree: provides diff. machine after files like
(iv) Virtual Hosting: hosting multiple domains on same server.
(v) Web proxy: sits b/w web client & web server to prevent IP blocking & maintain anonymity.

- Web server security issues:
Impacts: - Compromise of user accounts
- Website defacement
- Secondary attacks from website
- Root access to other applications on server
- Data tampering & data theft

NOTE: The Vulnerability stack for this is same as before.

- Enabling of unnecessary services
- Server installation with default settings
- Improper file & directory permissions
- Security conflicts with business case of use case
- Improper authentication with external system

- Web server famous attacks:
- DOS / DDOS (to prevent, captcha), DDOS → multiple IPs
- DNS server Hijacking
- DNS Amplification attack (similar to DDOS)
(Victim - ecommerce, etc)
- Directory Traversal attacks (../ requests to access restricted directory outside root directory)



- Man in the middle / Sniffing Attack

- Web Defacement: altering of visual appearance of a web page.

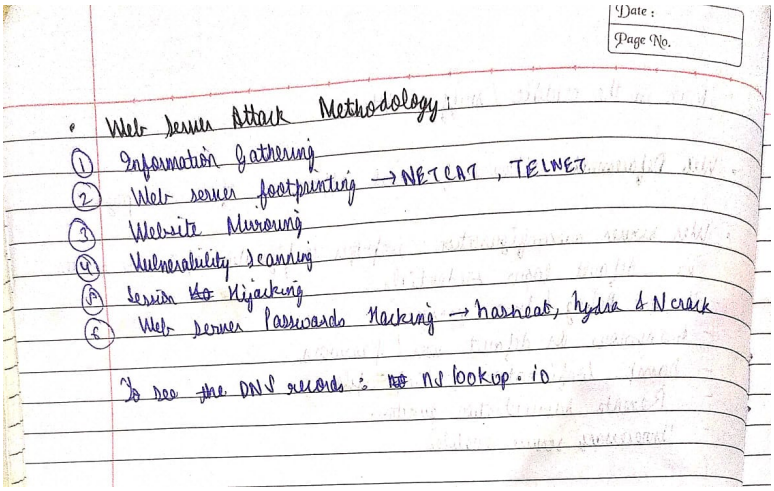
- Web server misconfiguration: improper configuration of a web server, ex: default login credentials.
- Verbose debug / Error messages
- Anonymous or default user / password.
- Sample configuration & script files
- Remote administration functions
- Unnecessary service enabled

- HTTP response - splitting attack.
- allows to inject additional HTTP header into a response, potentially leading to security risks such as cross-site scripting (XSS) attacks or cache poisoning.

- SSH Brute force attack.
- tries to guess password for an ssh account by using various combinations of usernames & passwords.

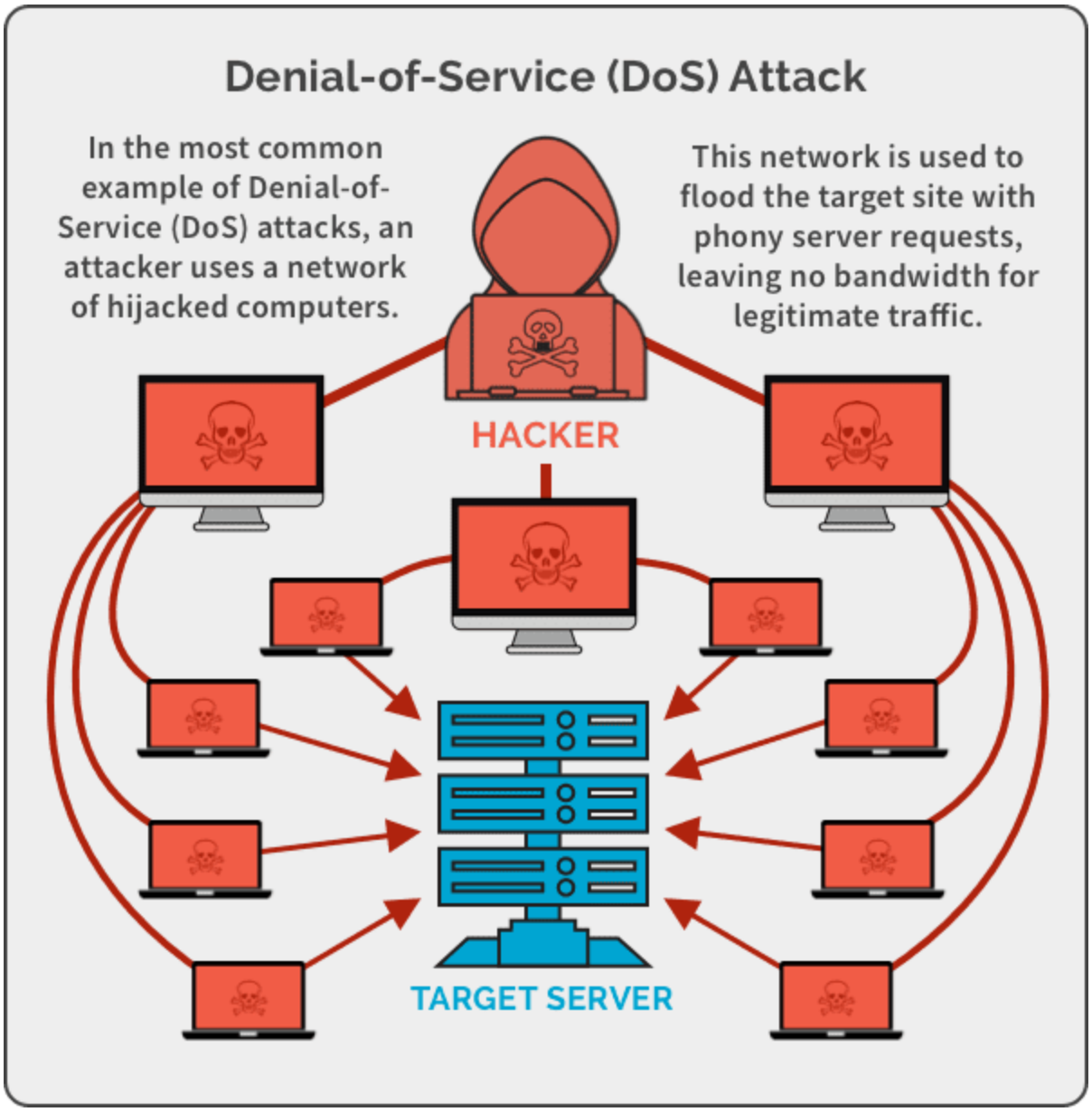
- Web server password cracking: similar to brute force attack.
- Server side Request Forgery Attack (SSRF): tricks a web application into sending unauthorized requests on behalf of the attacker, leading to unauthorized access to internal systems.

- Web Application Attacks:
- Parameter / Form Tampering
- Command Injection Attacks
- Source code Disclosure
- Buffer Overflow Attacks
- Cookie Tampering

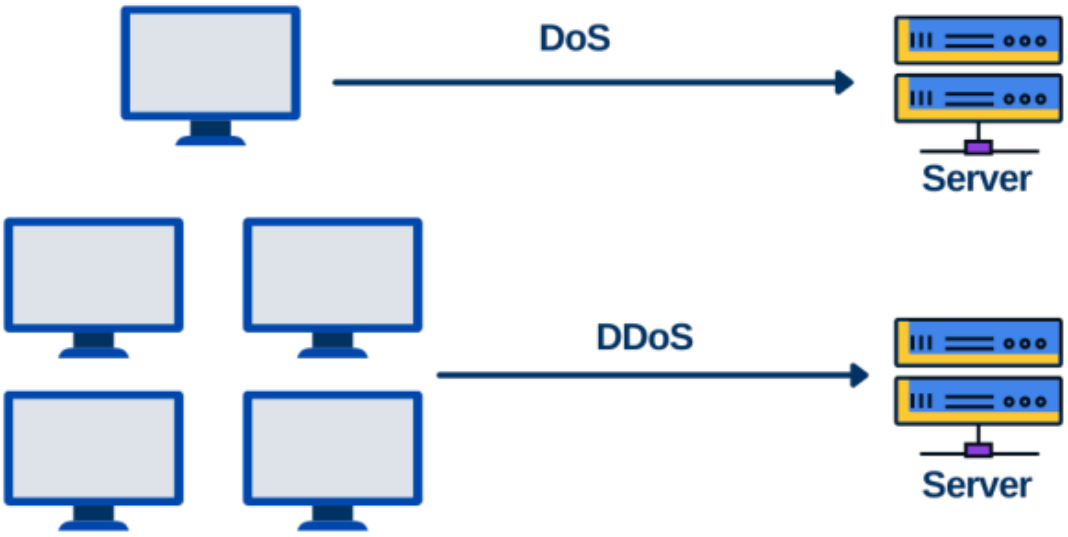


Task - 10 Web server based attacks

Denial-of-Service (DoS) Attack:
A DoS attack aims to overwhelm a web server's resources, making it unavailable to legitimate users. Attackers achieve this by sending an excessive amount of traffic, consuming network bandwidth or exhausting server resources such as CPU, memory, or disk space. This can be done by flooding the server with requests, overloading its resources, or exploiting a vulnerability in the server software.

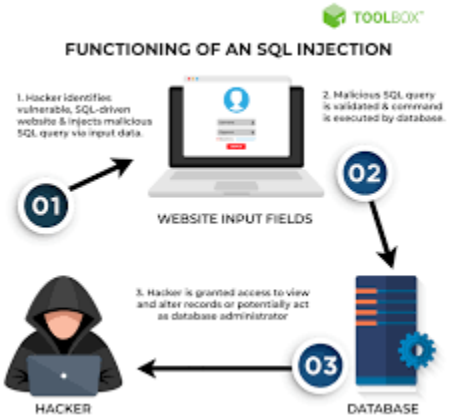


Distributed Denial-of-Service (DDoS) Attack:
Similar to a DoS attack, a DDoS attack involves multiple compromised computers (botnets) to flood a target server with traffic. This distributed nature makes it harder to defend against, as it increases the attack volume and diversity of sources.

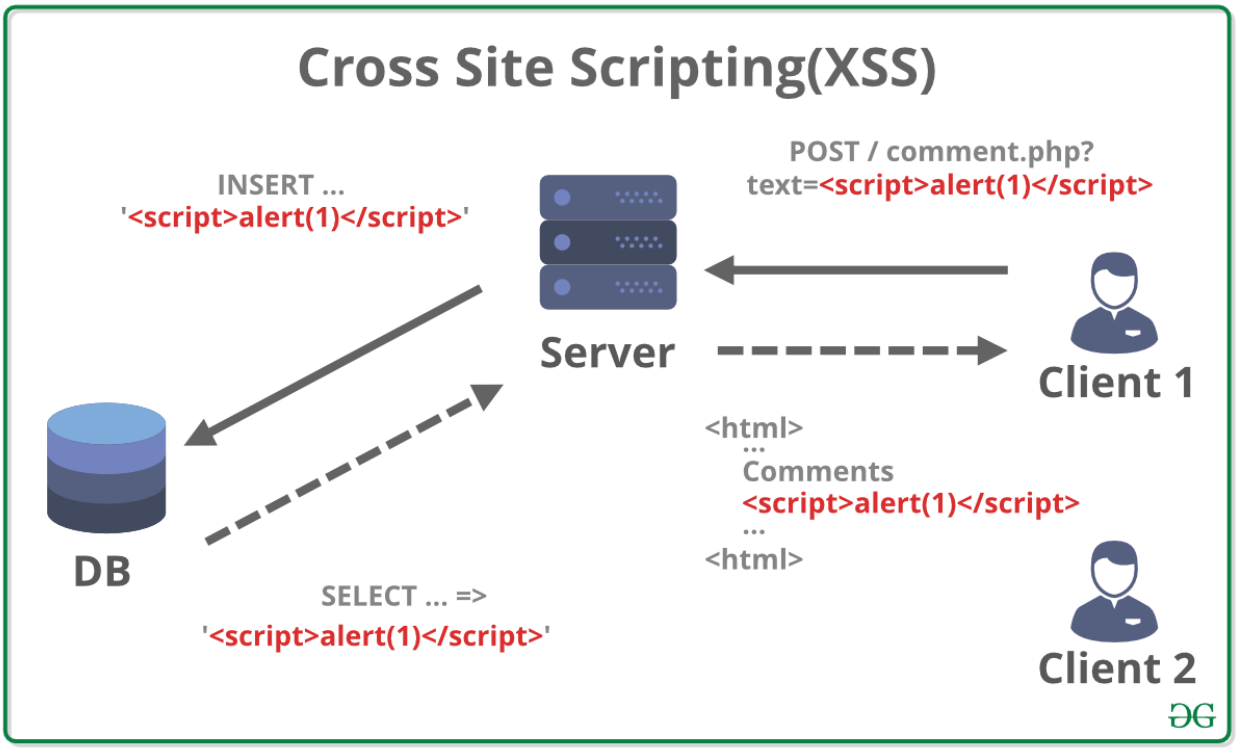


SQL Injection Attack:

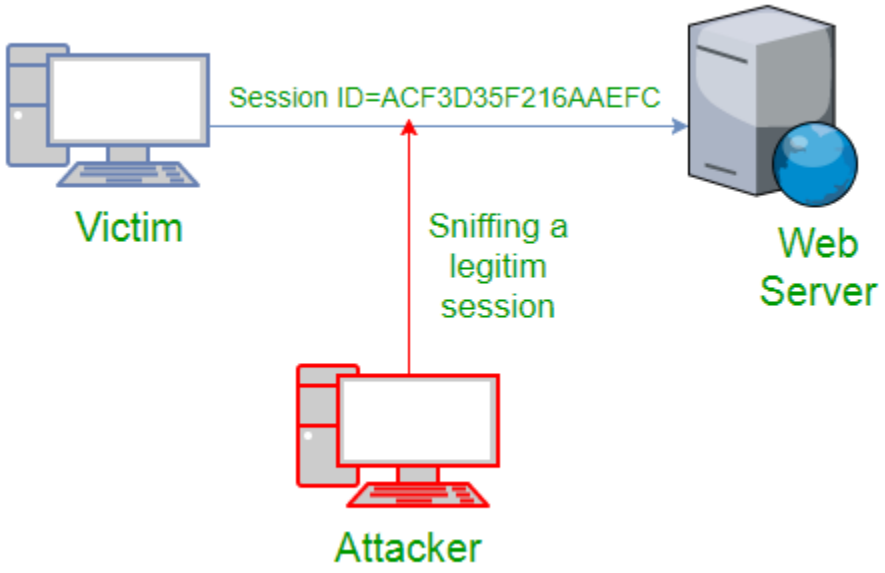
An SQL injection attack is a type of attack that exploits vulnerabilities in the SQL database used by a web application. In an SQL injection attack, attackers manipulate input fields to inject malicious SQL queries into a web application's database. If not properly sanitized, the application may execute these queries, potentially granting unauthorized access, data theft, or even control of the database.



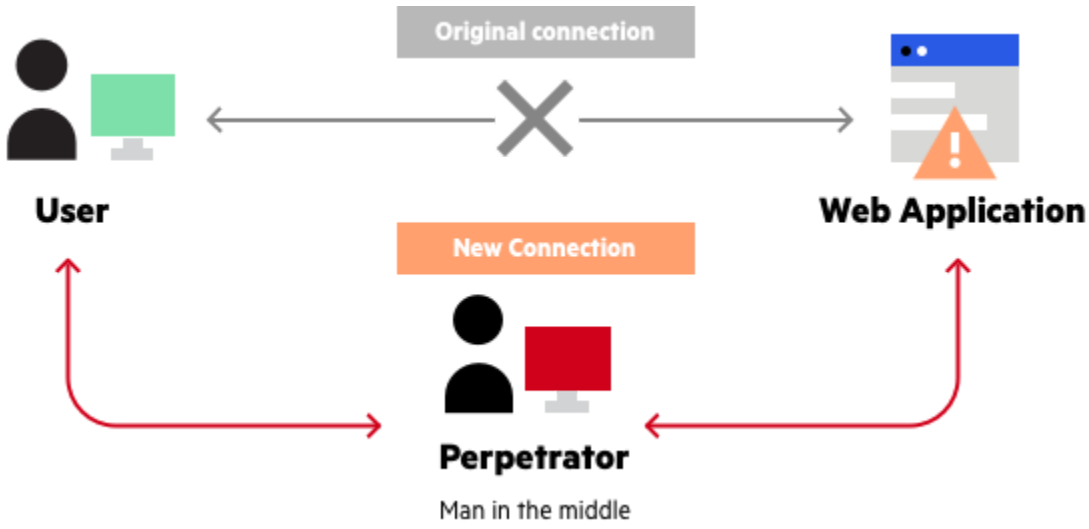
Cross-Site Scripting (XSS) Attack:
XSS attacks involve injecting malicious scripts into a web page, which then execute in the victim's browser. These scripts can steal user data, manipulate content, and hijack sessions, often exploiting vulnerabilities in user input validation. This code can then be executed by the victim's browser, which can steal cookies, hijack sessions, or display malicious content.



Session Hijacking:
Session hijacking involves stealing a user's session identifier, which grants the attacker access to the victim's authenticated session. This allows the attacker to impersonate the user and gain unauthorized access to their account.

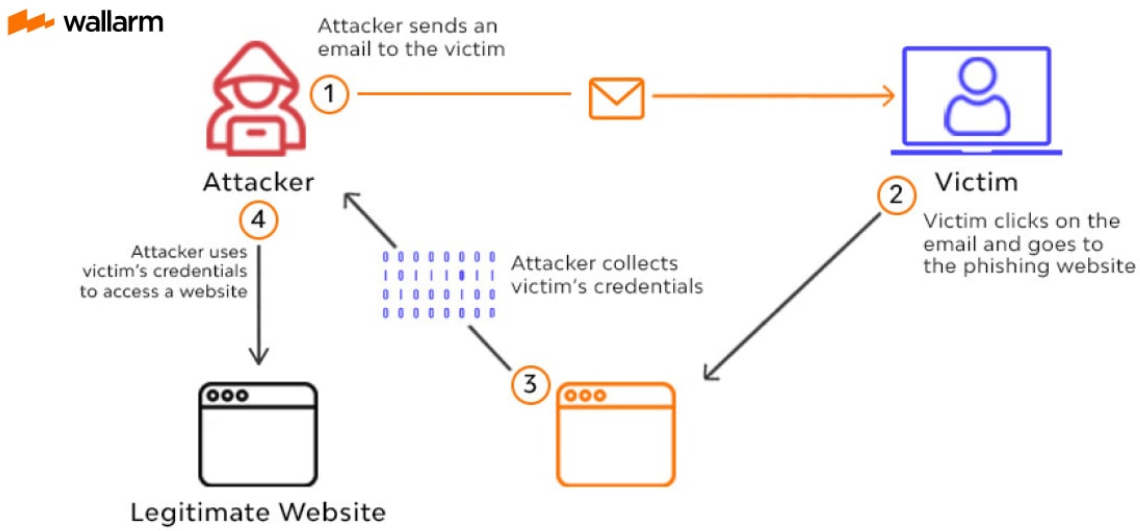


Man-in-the-Middle (MitM) Attack:
In a MitM attack, the attacker intercepts communication between two parties without their knowledge. This can expose sensitive data, like login credentials or payment details, and enable the attacker to modify or manipulate the communication.



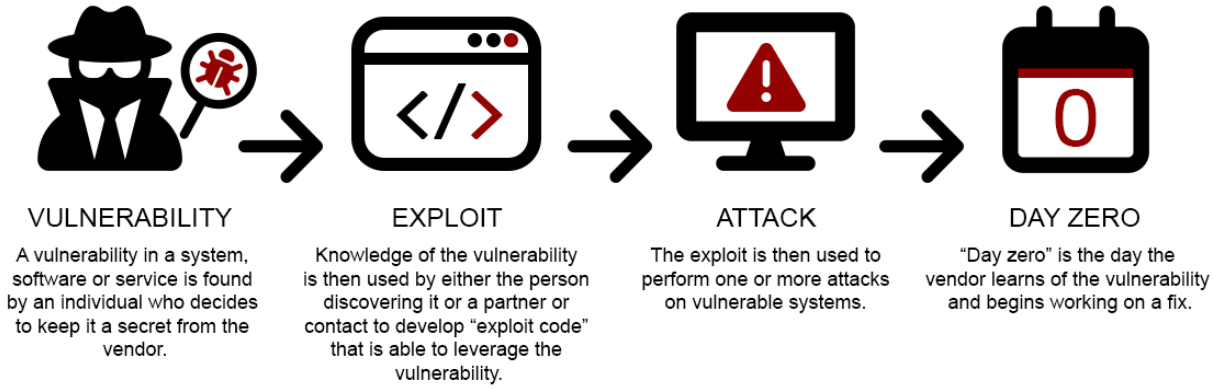
Phishing Attack:
Phishing attacks involve sending deceptive emails or messages to trick recipients into

clicking on malicious links or sharing sensitive information. These attacks often lead victims to fake websites that mimic legitimate ones. A phishing attack is an attack in which the attacker sends emails or text messages that appear to be from a legitimate source.

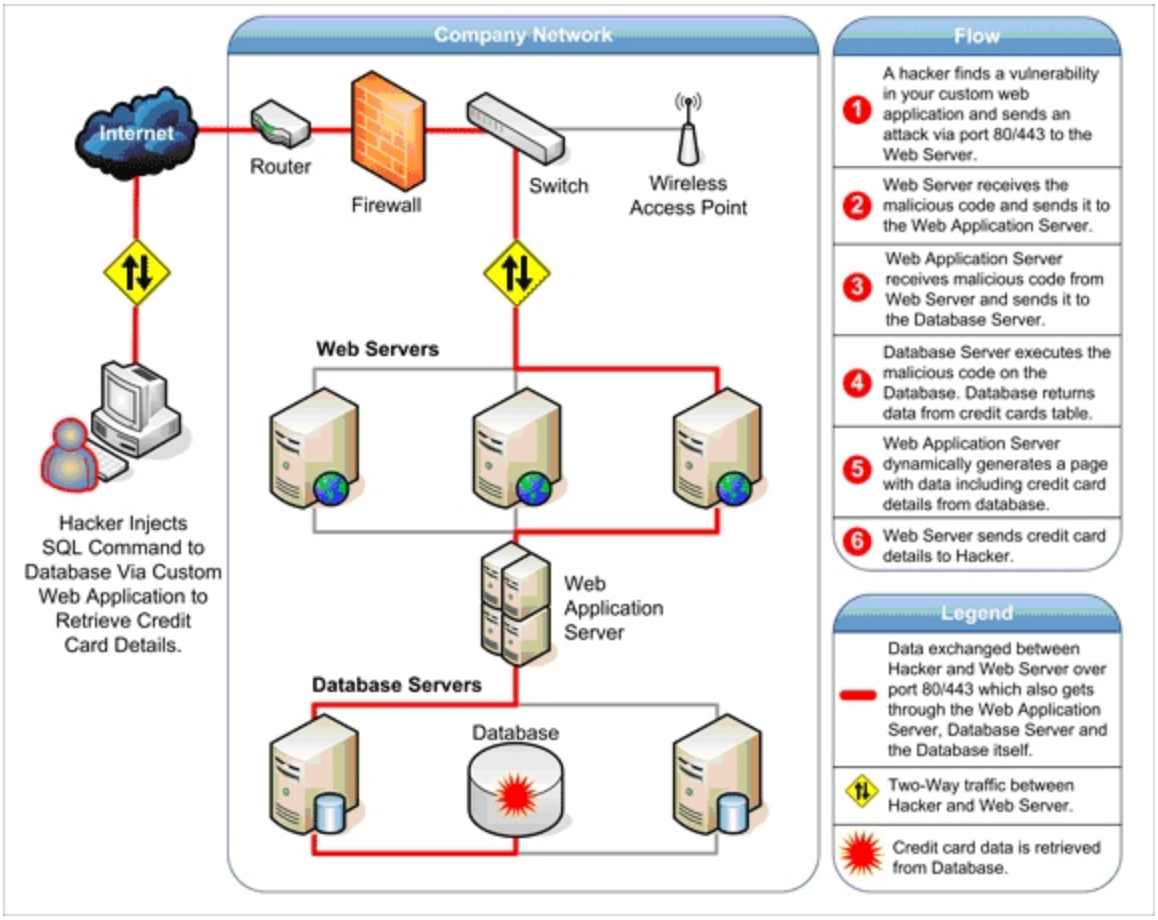


Malware Attack:
Malware attacks involve infecting a victim's device with malicious software, such as viruses, trojans, or ransomware. Malware can steal data, control the victim's system, or demand ransom payments for unlocking compromised data.

Zero-Day Attack:
A zero-day attack exploits vulnerabilities that are unknown to software vendors and, therefore, unpatched. Attackers use these vulnerabilities to gain unauthorized access to systems, steal data, or cause other harm. This makes it very difficult to defend against, as there is no patch available to fix the vulnerability.



Web Application Attack:
Web application attacks target vulnerabilities within the web application itself. These vulnerabilities can include coding errors, misconfigurations, or flaws in third-party libraries. Attackers exploit these weaknesses to compromise data, inject malicious code, or disrupt services.


Company Network
Internet
Router
Firewall
Switch
Wireless Access Point
Web Servers
Web Application Server
Database Servers
Database
Hacker Injects SQL Command to Database Via Custom Web Application to Retrieve Credit Card Details.
Flow
1 A hacker finds a vulnerability in your custom web application and sends an attack via port 80/443 to the Web Server.
2 Web Server receives the malicious code and sends it to the Web Application Server.
3 Web Application Server receives malicious code from Web Server and sends it to the Database Server.
4 Database Server executes the malicious code on the Database. Database returns data from credit cards table.
5 Web Application Server dynamically generates a page with data including credit card details from database.
6 Web Server sends credit card details to Hacker.
Legend
Data exchanged between Hacker and Web Server over port 80/443 which also gets through the Web Application Server, Database Server and the Database itself.
Two-Way traffic between Hacker and Web Server.
Credit card data is retrieved from Database.