Basics of hacking

INTRODUCTION

DURATION : 0’30

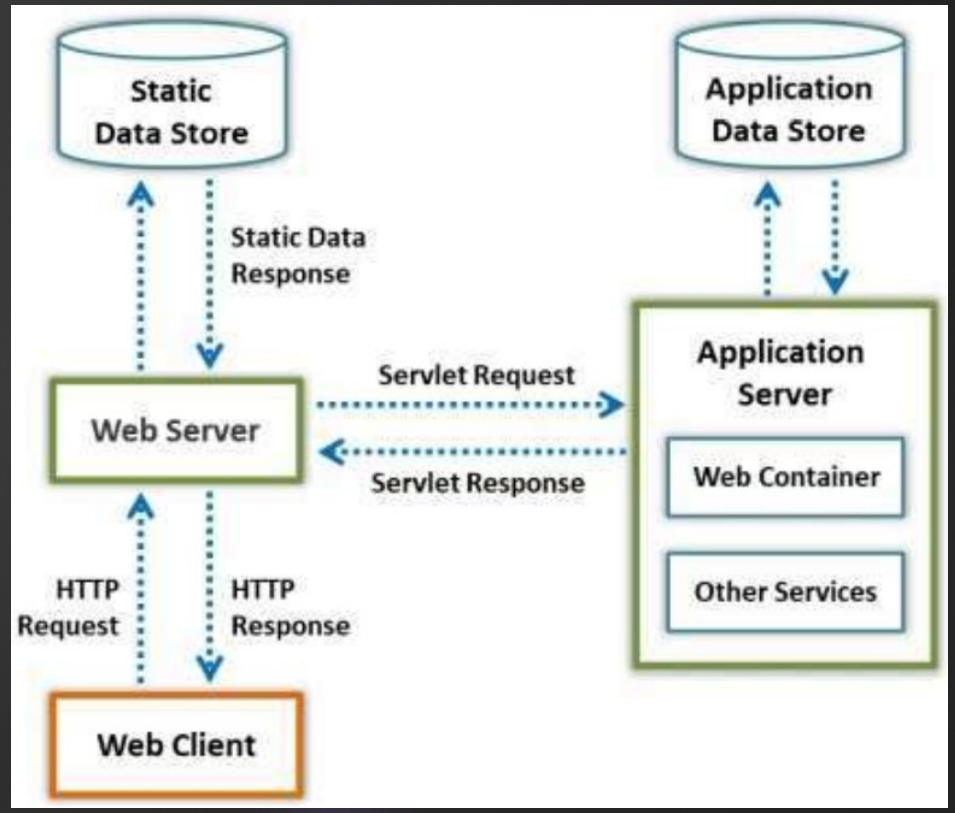
 Version : 0.0.8 Update : 10/03/2022

Module Objectives

- Web servers are a critical component of web infrastructure.
- A single vulnerability in web server configuration may lead to a security breach on websites.
- At the end oh this module, you will be able to do the following:
  - Describe web server concepts;
  - Perform various web server attacks;
  - Describe web server attack methodology;
  - Use different web server attack tools;
  - Apply web server attack countermeasures;
  - Describe patch management concepts;
  - Use different web server security tools.

Web Server Concepts

- A web server is a computer system that stores, processes, and delivers web pages to clients via the HTTP.
- In general, a client initiates a communication process through HTTP requests. When a client desires to access any resource such as web pages, photos, and videos, the client's browser generates an HTTP request that is sent to the web server.
- Depending on the request, the web server collects the requested information/content from the data storage or application servers and responds to the client's request with an appropriate HTTP response.
- If a web server cannot find the requested information, then it generates an error message.

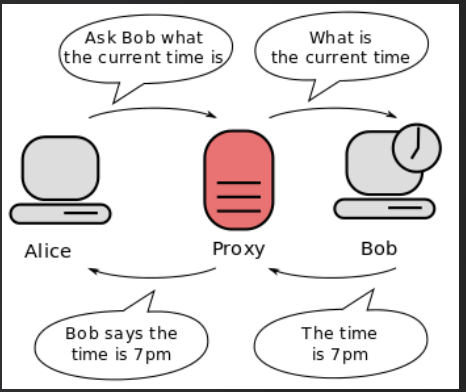


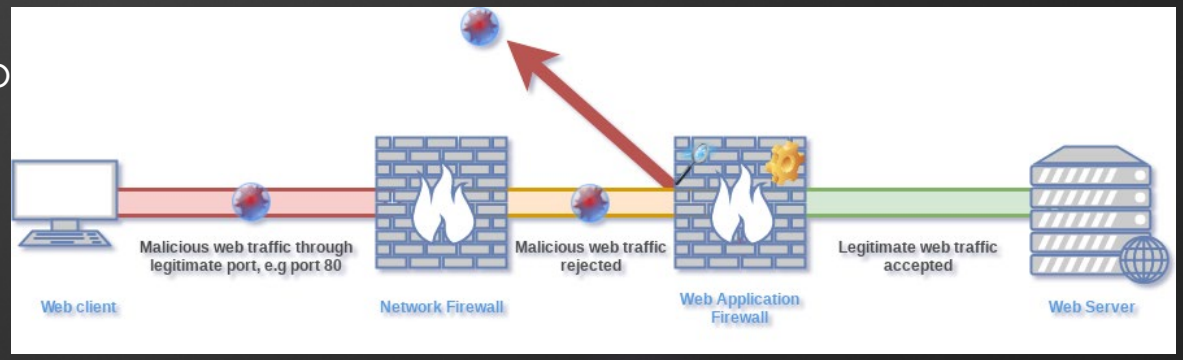
Components of a Web Server

- The **document root** is one of the root file directories of the web server that stores critical HTML files related to the web pages of a domain name, which will be sent in response to requests.
  If the requested URL is "*www.site1.com*" and the document root is named "*site1folder*" and is stored in the directory "*/admin/web*", then "*/admin/web/site1folder*" is the document directory address.
  If the complete request is "*www.site1.com/index.html*, the server will search for the file path "*/admin/web/site1/index.html*".
- **Server Root** is the top-level root directory under the directory tree in which the server's configuration and error, executable, and log files are stored. Often called "conf", "logs", and "cgi-bin".
- A **virtual document tree** provides storage on a same or different machine / disk. This allows for easy use of a huge number of virtual hosts with similar configurations.
- **Virtual Hosting :** It is a technique of hosting multiple domains or websites on the same server. This technique allows the sharing of resources among various servers. It is employed in large- scale companies.

Web security Components

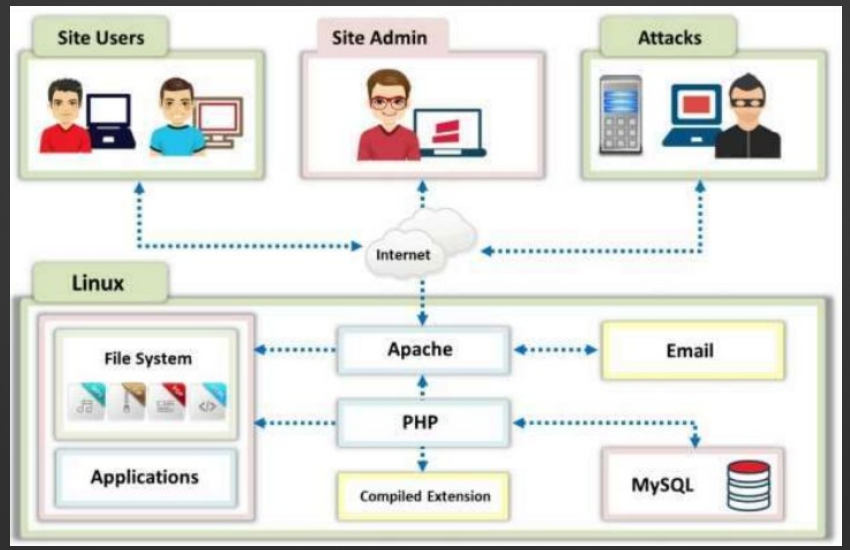
- A **proxy server** is located between the web client and web server. That acts as an intermediary by placing itself between two hosts principally to monitor their exchanges (allow / deny an IP, Domain, web category and can record all requests).
- Reverse Proxy + load balancer
- A **firewall** is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted network and an untrusted network, such as the Internet.
- statefull / stateless and IPS/IDS
- A **web application firewall (WAF)** filters, monito a web service. By inspecting HTTP traffic, it can prevent attacks exploiting a web application's known vulnerabilities, such as SQL injection, cross-site scripting (XSS), file inclusion, and improper system configuration.





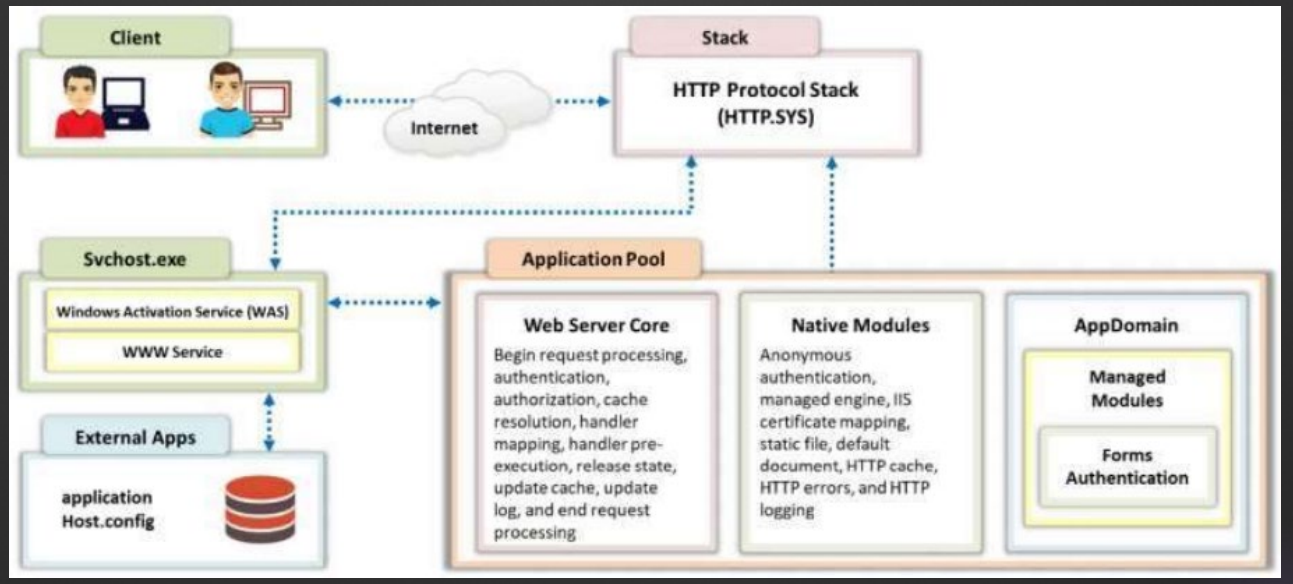
Open-source Web server architecture

- The following are the functions of the principal components in open-source web server architecture:
  - Linux is the operating system (OS) of the web server and provides a secure platform
  - Apache is the component of the web server that handles each HTTP request and response
  - MySQL is a relational database used to store the content and configuration information of the web server
  - PHP is the application layer technology used to generate dynamic web content



IIS Web server architecture

- Internet Information Service (IIS) is a web server application developed by Microsoft for Windows.
- IIS is a flexible, secure, and easy-to-manage web server for hosting anything on the web. It supports HTTP / HTTPS, File Transfer Protocol (FTP), FTP Secure (FTPS), Simple Mail Transfer Protocol (SMTP), and Network News Transfer Protocol (NNTP).
- It has several components, including a protocol listener such as HTTP.sys and services such as the World Wide Web Publishing Service and Windows Process Activation Service (WAS). Each component functions in application and web server roles. These functions may include listening to requests, managing processes, and reading configuration files.



Main problems on web servers

- A web server configured by poorly trained system administrators may have security vulnerabilities. Inadequate knowledge, negligence, laziness, and inattentiveness toward security can pose the greatest threats to web server security.
- The main problems are :
  - Failing to update the web server with the latest patches
  - Using the same system administrator credentials everywhere
  - Allowing unrestricted internal and outbound traffic
  - Running unhardened applications and servers

Impact of Web Server attacks

- **Compromise user accounts** to access registered user pages.
- **Website defacement** (change the appearance of a website).
- **Secondary attacks from the website** to launch further attacks on various websites or client systems.
- **Root access** to other applications or server.
- **Data tampering** by altering or deleting the data of a web server.
- **Data theft** (data are among the primary assets of an organization).
- **Damage reputation**.

Why and how Web Server are compromised ?

- The following are some oversights that can compromise a web server:
  - Improper file and directory permissions
  - Installing the server with default settings
  - Unnecessary services enabled / remote administration
  - Security conflicts with the business (ease-of-use requirements)
  - Lack of proper security policy, procedures, and maintenance
  - Default accounts with default or no passwords
  - Misconfigurations in the web server, OS, and networks
  - Do not apply patch or bugs in server software, OS, and web applications
  - Misconfigured encryption settings
  - Administrative or debugging functions that are enabled or accessible on web servers

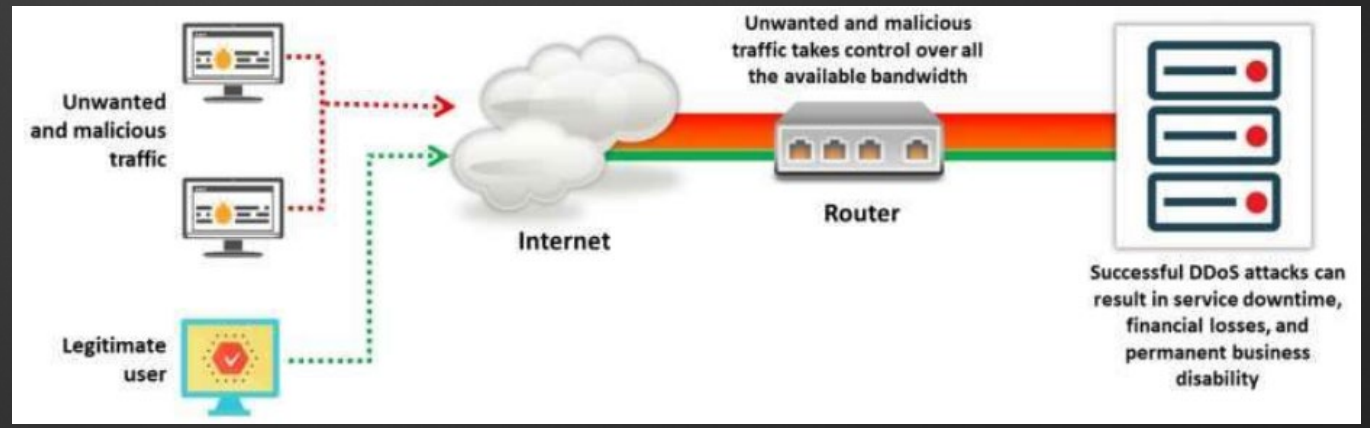
Web Server Attacks

Web Server attacks

- An attacker can use many techniques to compromise a web server, such as:
  - DoS/DDoS,
  - Domain Name System (DNS) server hijacking,
  - DNS amplification,
  - Directory traversal,
  - Man in the middle (MITM) / sniffing,
  - Phishing,
  - Website defacement,
  - Web server misconfiguration,
  - HTTP response splitting,
  - Web cache poisoning,
  - Secure Shell (SSH) brute force,
  - Web server password cracking,
  - Etc.

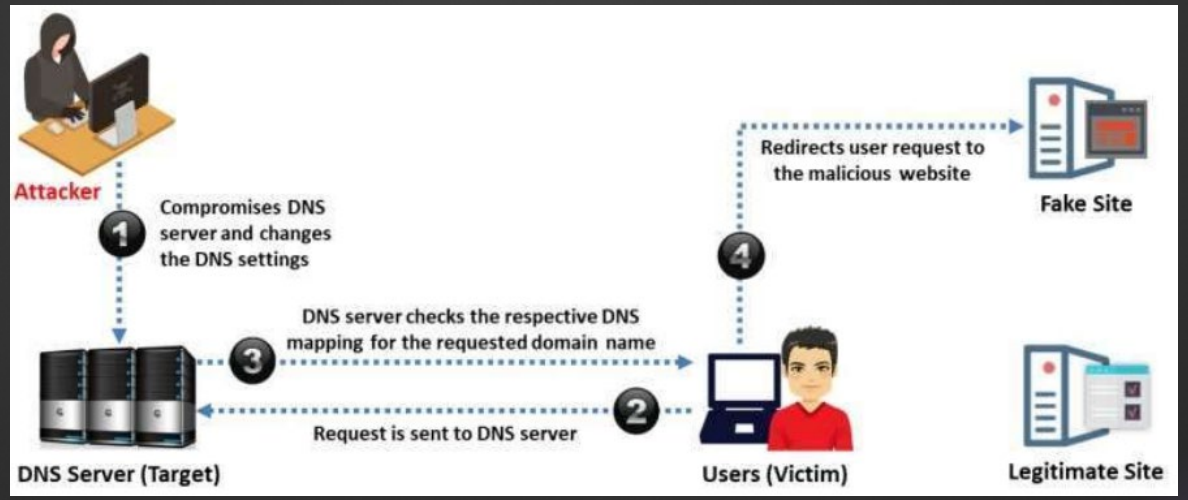
DoS/DDoS attack

- DoS/DDoS attack involves flooding targets with copious fake requests so that the target stops functioning and becomes unavailable to legitimate users.
- By using a web server DoS/DDoS attack, an attacker attempts to take the web server down or make it unavailable to legitimate users.
- To crash a web server running an application, the attacker targets the following services :
  - Network bandwidth
  - CPU usage
  - Server memory
  - Hard-disk space
  - Application exception handling
  - Database space mechanism



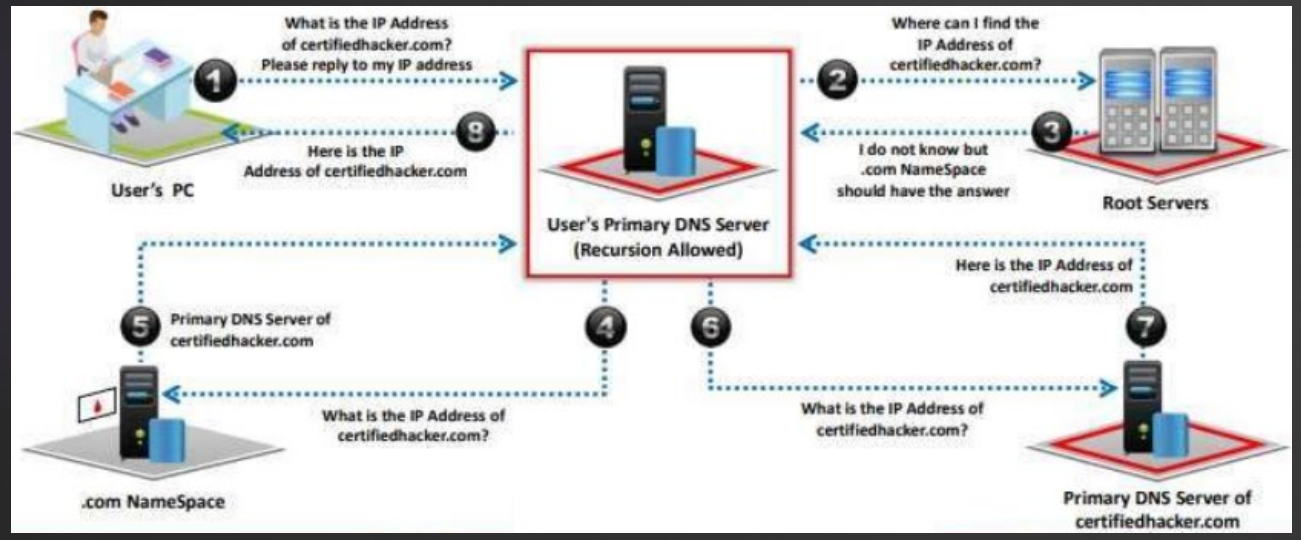
DNS Server Hijacking

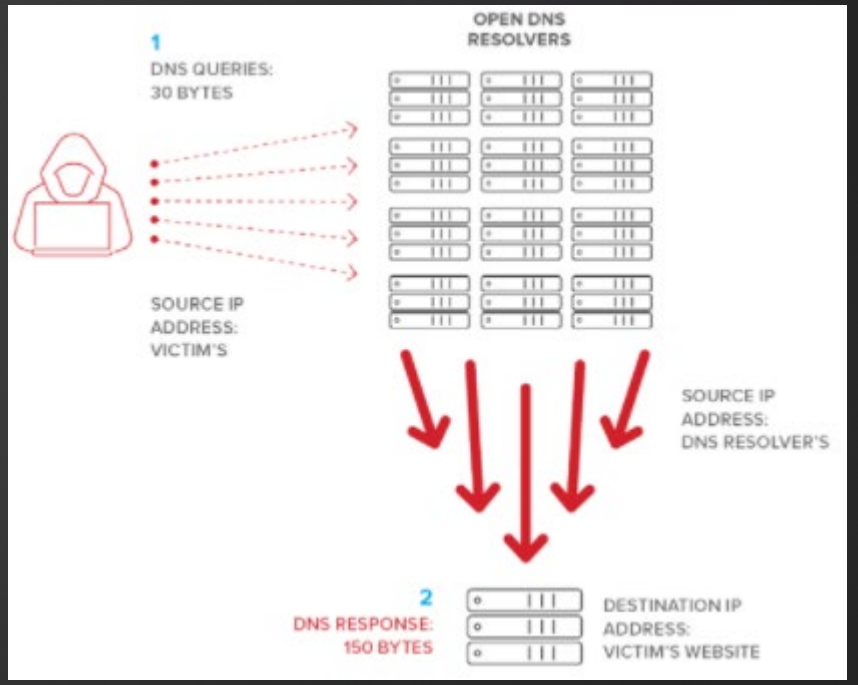
- The Domain Name System (DNS) resolves a domain name to its corresponding IP address. A user queries the DNS server with a domain name, and the DNS server responds with the corresponding IP address.
- In DNS server hijacking, an attacker compromises a DNS server and changes its mapping settings to redirect toward a rogue DNS server that would redirect the user's requests to the attacker's rogue server. Consequently, when the user enters a legitimate URL in a browser, the settings will redirect to the attacker's fake site.



DNS Amplification Attack

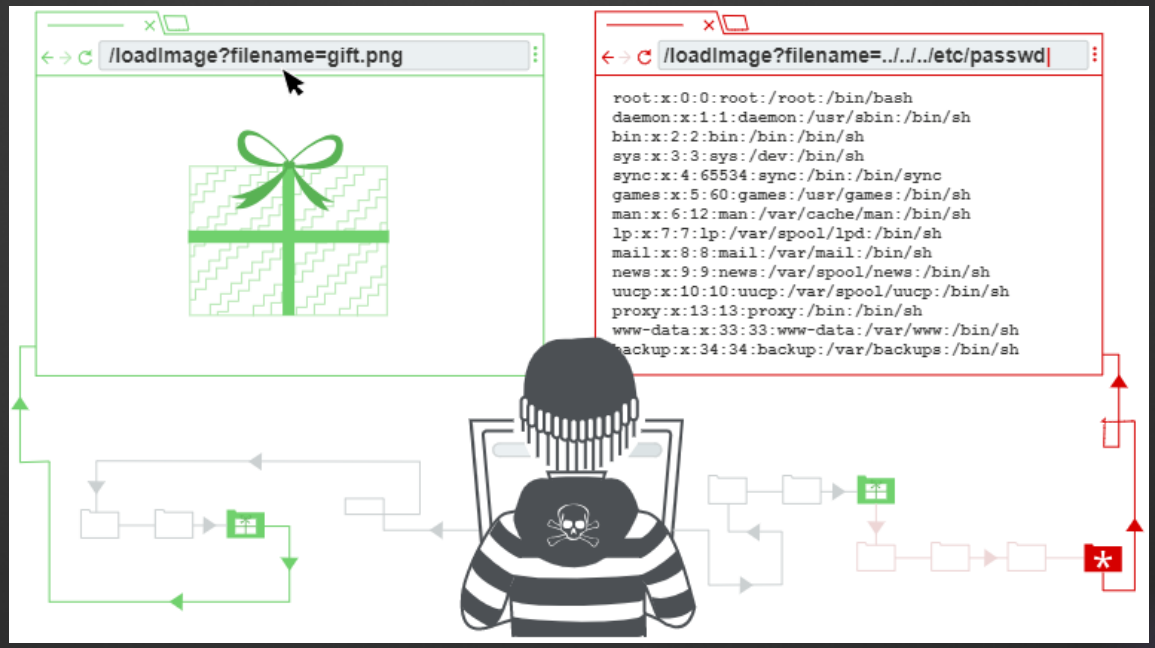
- Attacker takes advantage of the DNS recursive method of DNS redirection to perform DNS amplification attacks (left recursive and right not recursive).





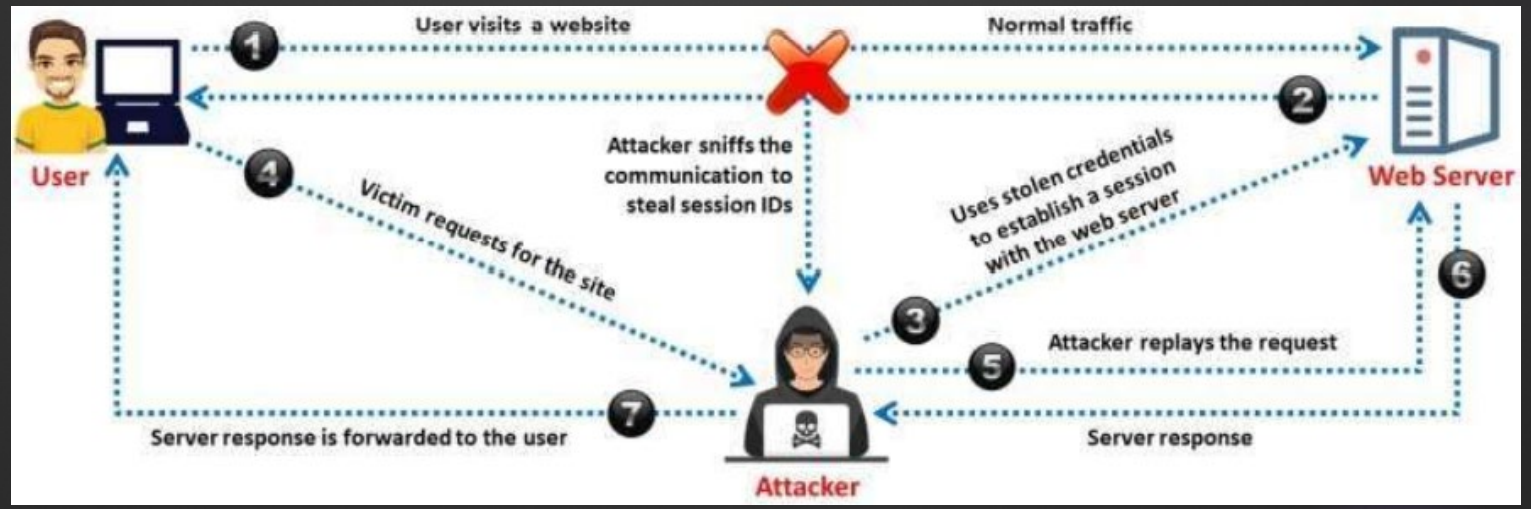
Local File Inclusion (LFI): Directory Path Traversal

- **Directory Path Traversal** (DPT) **is a part of** Local File Inclusion (**LFI**).
- **DPT** vulnerabilities **only allow an attacker to read a file**, while LFI and RFI may also allow an attacker to execute code and/or command.



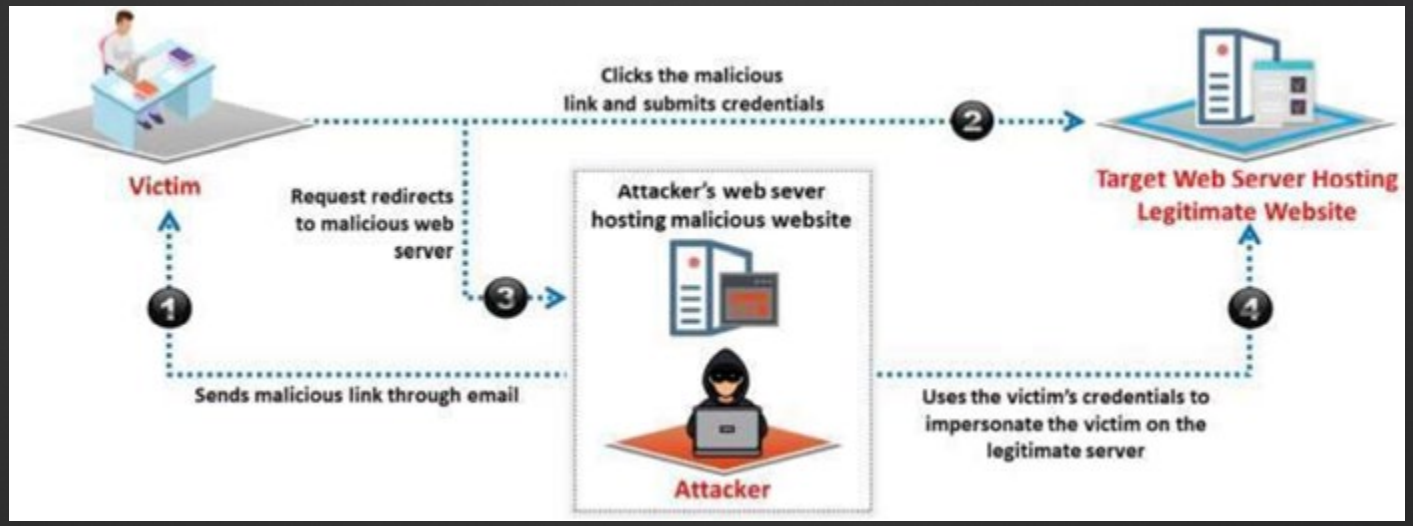
Man-in-the-Middle/Sniffing Attack

- Man-in-the-middle (MITM) attacks allow an attacker to access sensitive information by intercepting and altering communications between an end user and web servers.
- In an MITM attack or sniffing attack, an intruder intercepts or modifies the messages exchanged between the user and web server by eavesdropping or intruding into a connection.
- This allows an attacker to steal sensitive user information, such as online banking details, usernames and passwords, etc.
- The attacker lures the victim to connect to the web server like a proxy.



Phishing Attacks

- The attacker tricks the user to submit login details for a website that looks legitimate, and redirects them to the malicious website hosted on the attacker's web server.
- The attacker then steals the credentials entered and uses them to impersonate the user with the website hosted on the legitimate target server.
- Attacker can then perform unauthorized or malicious operations on the target legitimate website.



Website Defacement

- Web defacement occurs when an intruder maliciously alters the visual appearance of a web page by inserting or substituting provocative, and frequently, offending data.
- Defaced pages expose visitors to some propaganda or misleading information until the unauthorized changes are discovered and corrected
- Attackers use a variety of methods such as MTSQL injection to access a site In order to deface it

Web Server Misconfiguration

- Server misconfiguration refers to configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft. Example :
  - Web Server Misconfiguration
  - Verbose Debug/Error Messages
  - Anonymous or Default Users/Passwords
  - Sample Configuration and Script Files
  - Remote Administration Functions
  - Unnecessary Services Enabled

httpd.conf : this conf. allows anyone to view the server status page including current hosts.

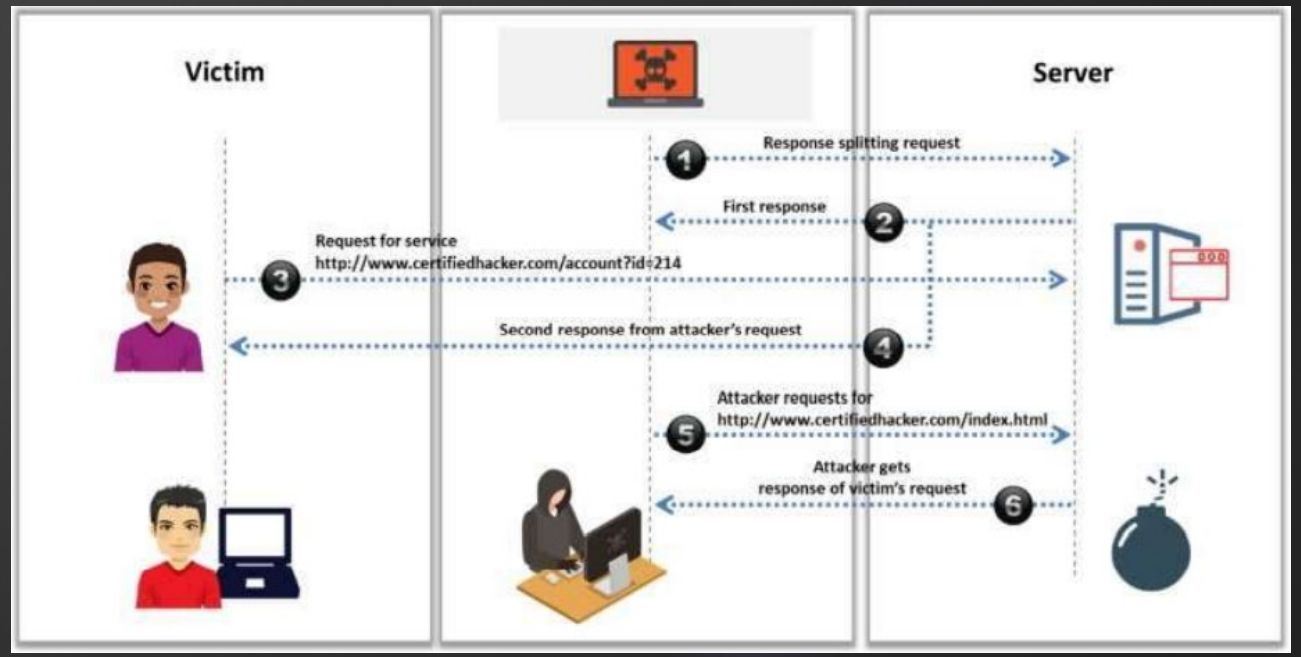
```
<Location /server-status>
SetHandler server-status
</Location>
```

php.ini : this conf. generates verbose error messages

```
display_error = On
log_errors = On
error_log = syslog
ignore_repeated_errors = Off
```

HTTP Response-Splitting Attack 1/2

- The attacker sends a response-splitting request to the web server.
- The server splits the response into two and sends the first response to the attacker and the second response to the victim.
- After receiving the response from the web server, the victim requests service by providing credentials.
- Simultaneously, the attacker requests for the index page.
- Subsequently, the web server sends the response to the victim's request to the attacker, and the victim remains uninformed.



More information : HTTP Response Splitting Software Attack | OWASP Foundation

HTTP Response-Splitting Attack 2/2

- The following code segment reads the name of the author from an HTTP request and sets it in a cookie header of an HTTP response.

```
String author = request.getParameter(AUTHOR_PARAM);
...
Cookie cookie = new Cookie("author", author);
    cookie.setMaxAge(cookieExpiration);
    response.addCookie(cookie);
```

- Assuming a string consisting of alpha-numeric characters, such as "Jane Smith", is submitted in the request the HTTP response including this cookie might take the following form:

```
HTTP/1.1 200 OK
...
Set-Cookie: author=Jane Smith
...
```

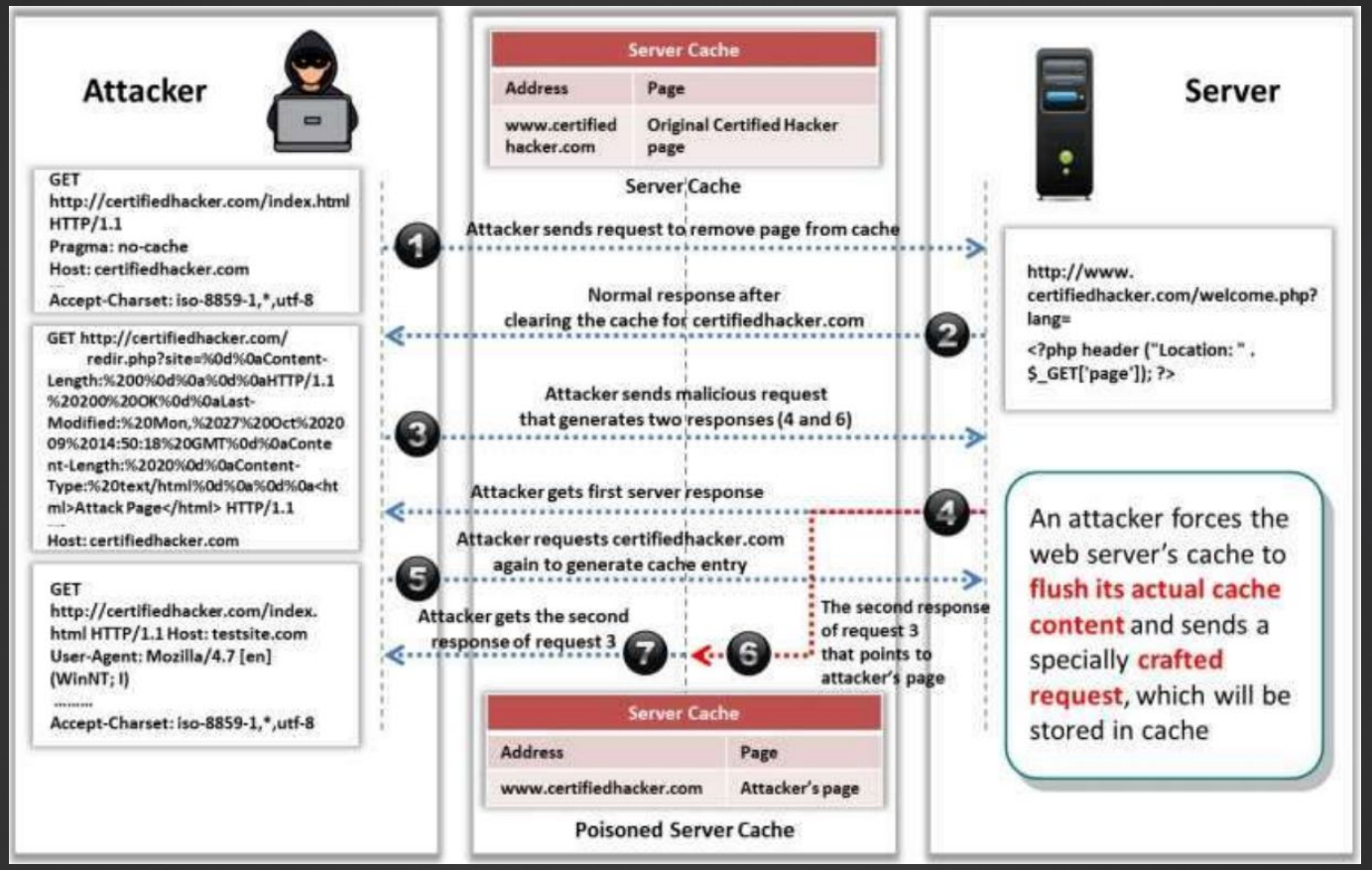
- However, because the value of the cookie is formed of unvalidated user input, the response will only maintain this form if the value submitted for AUTHOR_PARAM doesn't contain any CR and LF characters. If an attacker submits a malicious string, such as "Wiley Hacker\r\nContent-Length:45\r\n\r\n…", then the HTTP response would be split into an imposter response followed by the original response, which is now ignored:

```
HTTP/1.1 200 OK
...
Set-Cookie: author=Wiley Hacker
Content-Length: 999

<html>malicious content...</html> (to 999th character in this example)
Original content starting with character 1000, which is now ignored by the web browser...
```

- This type of attack exploits vulnerabilities in input validation like Cross-site scripting (XSS), cross-site request forgery (CSRF), and Structured Query Language injection (SQLi).

Web Cache Poisoning Attack



- A poisoned web cache can potentially be a devastating means of distributing numerous different attacks, exploiting vulnerabilities such as XSS, JavaScript injection, open redirection, and so on.
- A problem is the length of the URI, which sometime makes it impossible to put the necessary response header, which would next be matched to the request for the poisoned page.

SSH Brute Force Attack

- SSH protocols are used to create an encrypted SSH tunnel between two hosts to transfer data over a network.
- Attackers can brute force SSH login credentials to gain unauthorized access to an SSH tunnel.
- SSH tunnels can be used to transmit malware and other exploits to victims without being detected.

Web Server Password Cracking

- Passwords can be cracked manually or by guessing or by performing dictionary, brute force, and hybrid attacks using automated tools such as THC Hydra, Ncrack, RainbowCrack, etc.
- Attackers use different methods such as social engineering, spoofing, phishing, using a Trojan Horse or virus, wiretapping, and keystroke logging (keylogger).
- Attacker mainly targets SMTP servers, web shares, ssh tunnels, web form, ftp servers, etc.
- Techniques:
  - Attacker guesses possible passwords (admin, password, etc.);
  - Brute-force with a dictionary;
  - Standard brute-force (A to Z, 0 to 9, etc.);
  - Hybrid attack use both dictionnary and brute-force.

Server-Side Request Forgery (SSRF)

- Attackers exploit SSRF vulnerabilities in a public web server to send crafted requests to the internal or back end servers.
- Once the attack is successfully performed, the attackers can perform various activities such as port scanning, network scanning, IP address discovery, reading web server files, and bypassing host-based authentication.

Two examples (against the server itself and against other back-end systems):
1. This causes the server to make a request to the specified URL, retrieve the stock status, and return this to the user.
2. An attacker can modify the request to specify a URL local to the server itself.

```
POST /product/stock HTTP/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 118

stockApi=http://stock.weliketoshop.net:8080/product/stock/check%3FproductId%3
D6%26storeId%3D1
```

```
POST /product/stock HTTP/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 118

stockApi=http://localhost/admin
```

```
POST /product/stock HTTP/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 118

stockApi=http://192.168.0.68/admin
```