**Hacking Webservers**

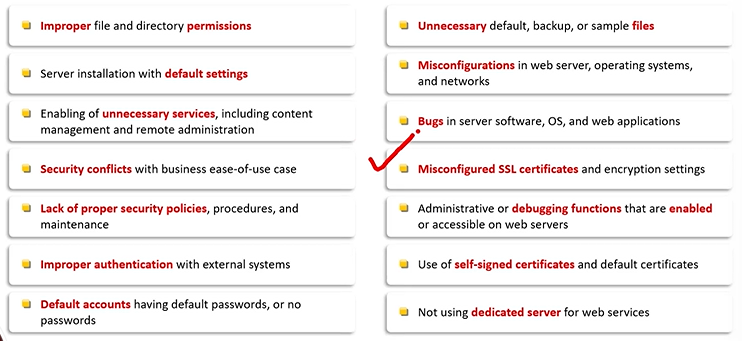
**What Are Web Servers?**

A web server is software or hardware that stores, processes, and delivers web content (HTML files, images, etc.) to users over the internet using HTTP or HTTPS. It handles incoming requests from clients (browsers) and returns the requested resources, enabling access to websites.

**Components of a Web Server:**

* **Web Server Software**: Handles HTTP requests and responses (e.g., Apache, Nginx).
* **Operating System**: Manages system resources and server operations (e.g., Linux, Windows).
* **Document Root**: Directory where web content is stored.
* **File System**: Retrieves files from storage for delivery.
* **Log Files**: Tracks server activity (access and error logs).
* **Dynamic Content Handlers**: Processes scripts to generate dynamic content (e.g., PHP, CGI).
* **Security Components**: Ensures encrypted communication and server security (e.g., SSL/TLS).
* **Application Server**: Runs server-side applications for dynamic content generation (e.g., Node.js, Ruby on Rails).
* **Content Management System (CMS)**: Manages content and interactions (e.g., WordPress).

**Why are web servers compromised?**

****

**Directory Traversal Attacks:**

Directory traversal attacks exploit vulnerabilities in web applications, allowing attackers to access restricted files or directories outside the intended web directory. By manipulating file paths (e.g., using ../ sequences), attackers can gain unauthorized access to sensitive files, such as configuration files, passwords, or system files, leading to potential exploitation.

**Website Defacement:**

Website defacement is a cyberattack where attackers gain unauthorized access to a website and alter its content, usually replacing it with their own messages or graphics. This typically targets the site's homepage, aiming to damage its reputation, spread propaganda, or demonstrate security vulnerabilities, without causing permanent data loss.

**Web Server Misconfiguration:**

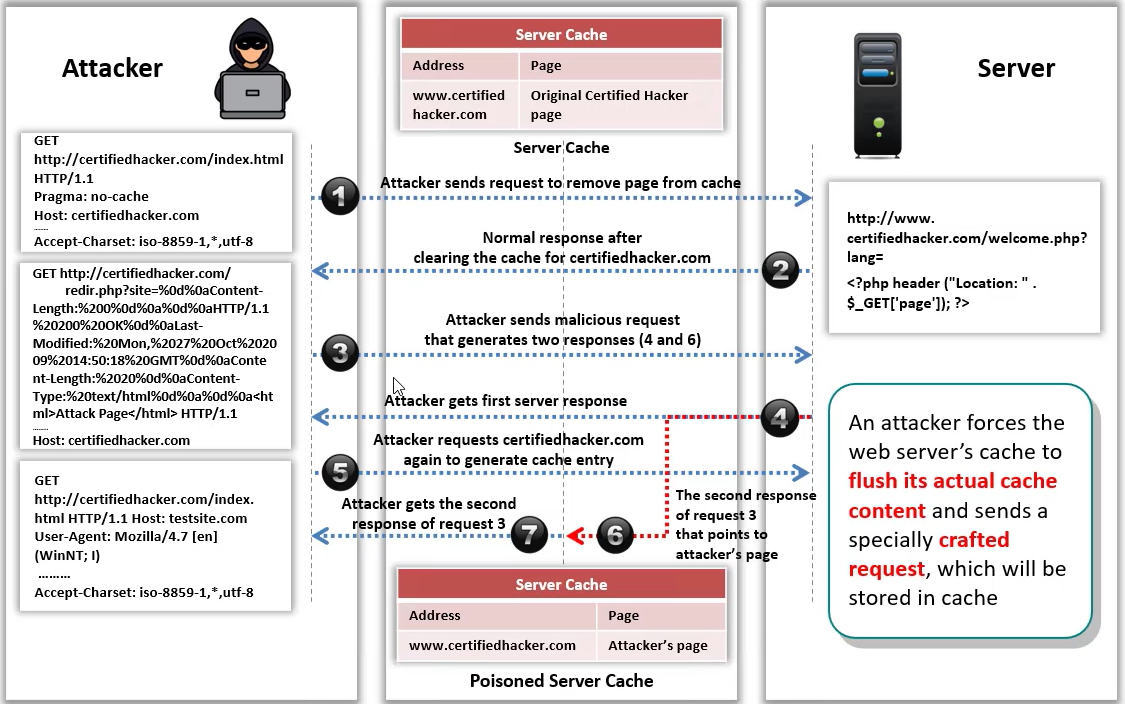
Server misconfiguration refers to configuration weakness in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft.

**HTTP response-splitting attack:**

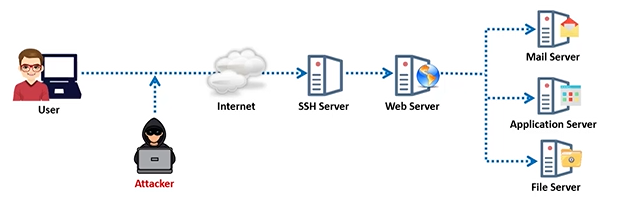
1. HTTP response splitting attack involves adding header response data into the input filed so that the server splits the response into two responses.
2. The attacker can control the first response to redirect the user to a malicious website whereas the other responses are discarded by the web browser.

**Web cache poisoning attack:**

Web cache poisoning is an attack where malicious data is injected into a web cache, causing it to serve harmful or altered content to users. By manipulating cacheable responses, attackers can deliver incorrect or malicious data (e.g., fake login pages), compromising user security and disrupting website functionality.



**SSH Brute Force attack:**



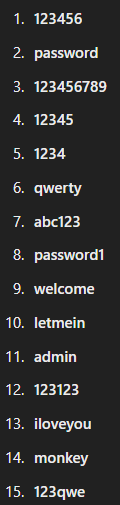
An SSH brute force attack is an attempt to gain unauthorized access to a server by systematically guessing SSH login credentials (username and password). Attackers use automated tools to try numerous combinations, exploiting weak or common passwords. If successful, they can compromise the server and execute malicious actions.

**Web Server Password Cracking:**

Web server password cracking involves attempting to guess or decrypt passwords used to authenticate access to a web server. Attackers use techniques like brute force, dictionary attacks, or exploiting weak passwords to gain unauthorized access. Once successful, they can manipulate server settings, steal data, or launch further attacks.

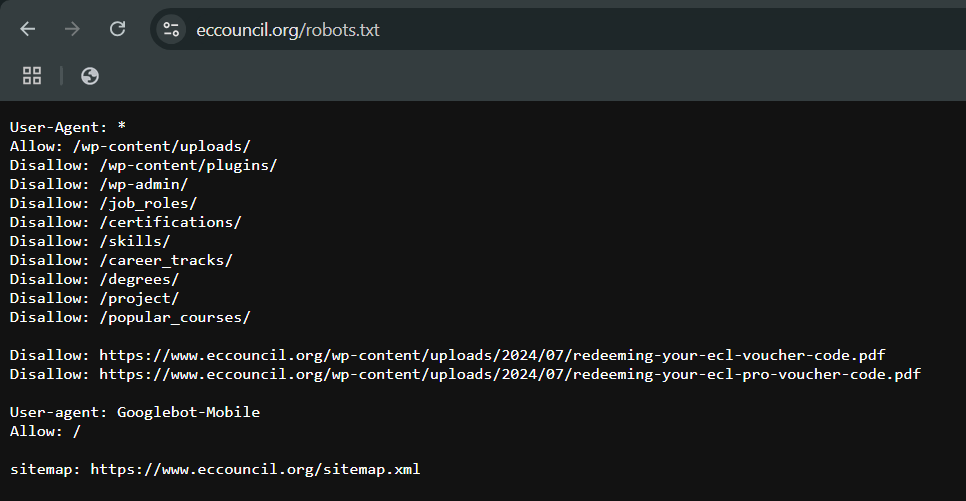
Attackers mainly targets: SMTP servers, FTP servers, SSH Tunnels, Web Shares etc.

Some common passwords we uses are as follows:



**Web Server Attack Methodology:**

1. Information gathering
2. Webserver Foot printing
3. Website Mirroring
4. Vulnerability Scanning
5. Session Hijacking
6. Web server Passwords hacking

**Information Gathering From Robots.txt file:**

The **robots.txt** file is a publicly accessible file placed on a website's root directory to provide instructions to web crawlers (also known as bots or spiders) about which pages or sections of the site they are allowed or disallowed to access and index.

**Information Gathering From robots.txt:**

By examining a website's **robots.txt** file, attackers or security researchers can gather useful information about the website's structure and potential vulnerabilities. Here’s how:

1. **Disallowed Directories/Files**: The robots.txt file may list directories or files that are restricted from being crawled, such as admin panels, backups, or sensitive files. While these areas are blocked for search engines, they may indicate potential targets for attackers.
   1. Example: Disallow: /admin/
2. **Sensitive Paths**: Certain paths may provide clues about the website’s infrastructure or security weaknesses (e.g., /backup/, /dev/, /config/). These could potentially lead to discovering important files or pages that might not be secured properly.
3. **Hidden Pages or Resources**: Sometimes, webmasters mistakenly leave sensitive pages or resources exposed through the robots.txt file, unintentionally hinting at places that should have been restricted or secured.
4. **User-Agent Details**: The file specifies which user agents (bots) are allowed or disallowed. Understanding which bots are given access can help attackers identify which bots they can use or manipulate for specific actions (e.g., scraping).
5. **Site Map Location**: The file can contain a link to the website's XML sitemap (Sitemap: http://example.com/sitemap.xml). This can provide attackers with a map of all indexed pages, potentially revealing hidden or overlooked areas of the site.

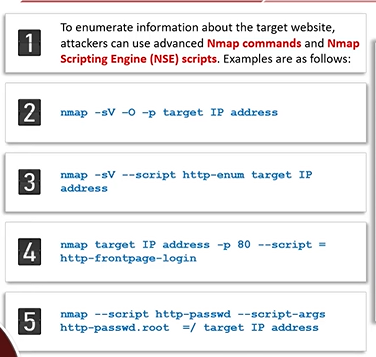
**How Attackers Use This Information:**

* **Finding Hidden Directories**: An attacker may look for disallowed directories and try to access them directly to find exposed or unprotected files.
* **Locating Sensitive Pages**: Attackers can try to access files like backup files or configuration files that could contain sensitive information (e.g., usernames, passwords).
* **Scraping the Site**: An attacker may scrape publicly accessible parts of the site to look for vulnerabilities or extract data for further attacks.

**Protection:**

* **Limit Sensitive Data in robots.txt**: Do not include any information in the robots.txt file that could expose sensitive data or structure.
* **Secure Disallowed Resources**: Just because a file or directory is disallowed in robots.txt does not mean it’s secure; ensure proper access controls are in place.

**Enumerating Web Server Information using Nmap:**



**Finding Default Content of Web Server:**

Tools used: Nikto2

Finding default content on a web server refers to identifying standard files or pages that come with web server software installations, such as Apache, Nginx, or IIS. These often include default pages (e.g., "index.html", "default.php") or error pages that can reveal server details, version information, or misconfigurations.

**Patches and Hotfixes:**

**Patches**:  
Patches are updates released by software vendors to fix bugs, security vulnerabilities, or performance issues in an application or system. They are typically installed to enhance functionality and address known problems. Patches are essential for maintaining software security and stability.

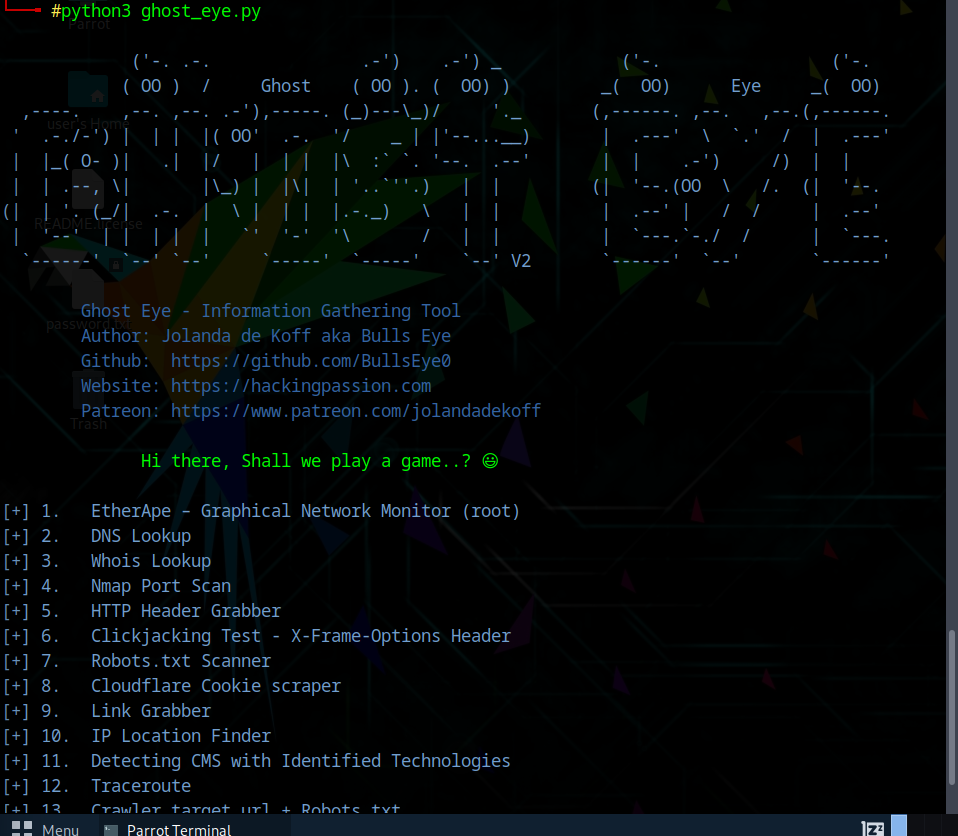
**Hotfixes**:  
Hotfixes are urgent, targeted updates designed to fix specific issues or vulnerabilities in software, often addressing critical problems that require immediate attention. Unlike regular patches, hotfixes are typically released outside the regular update cycle and are deployed to resolve high-priority issues quickly.

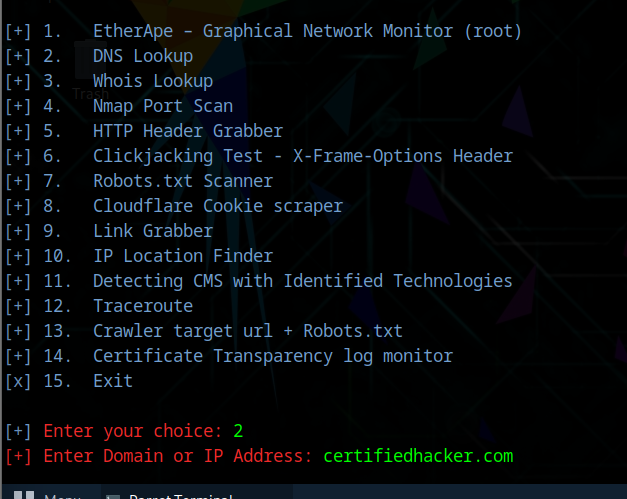
**What is Patch Management?**

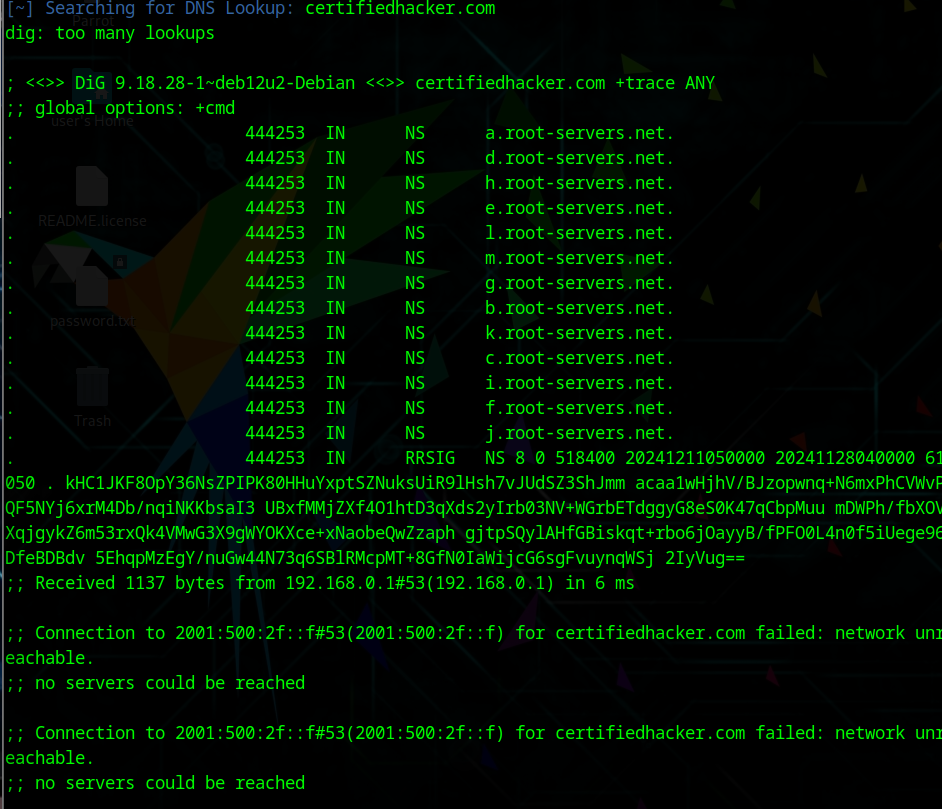
It is a process used to fix known vulnerabilities by ensuring that the appropriate patches are installed on a system.

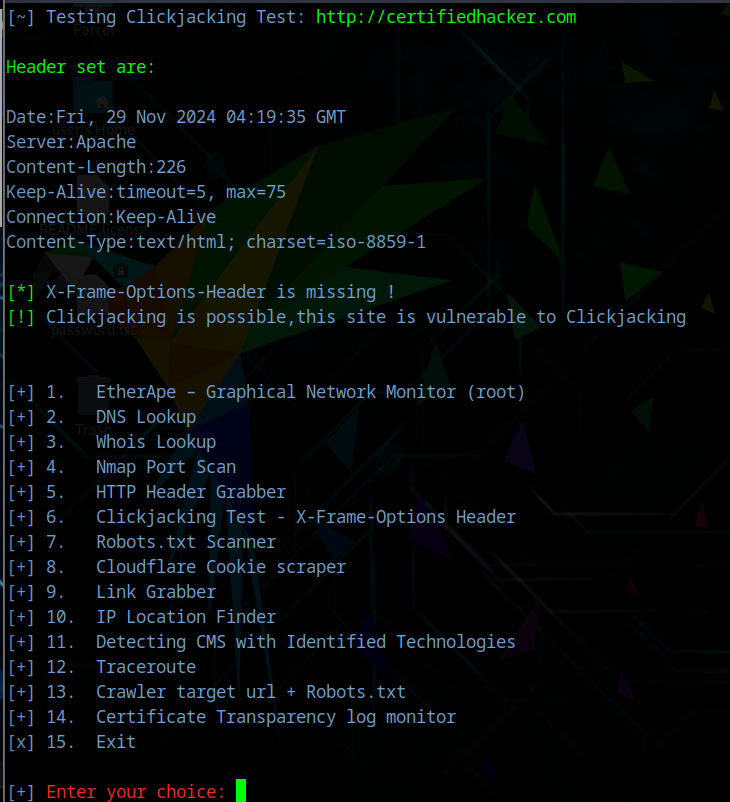
Tools: GFI languard, Solarwinds patch Manager, Software vulnerability manager etc.

**Information gathering through Ghost EYE:**

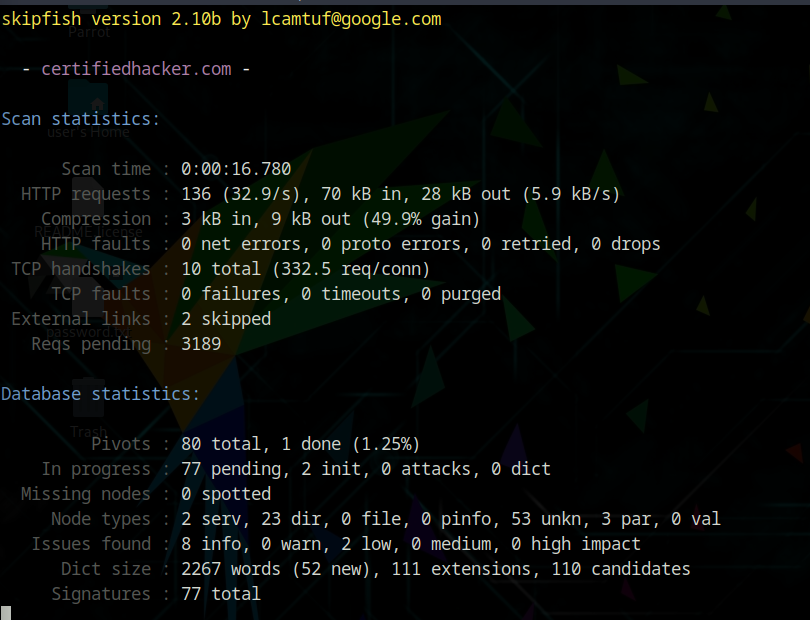


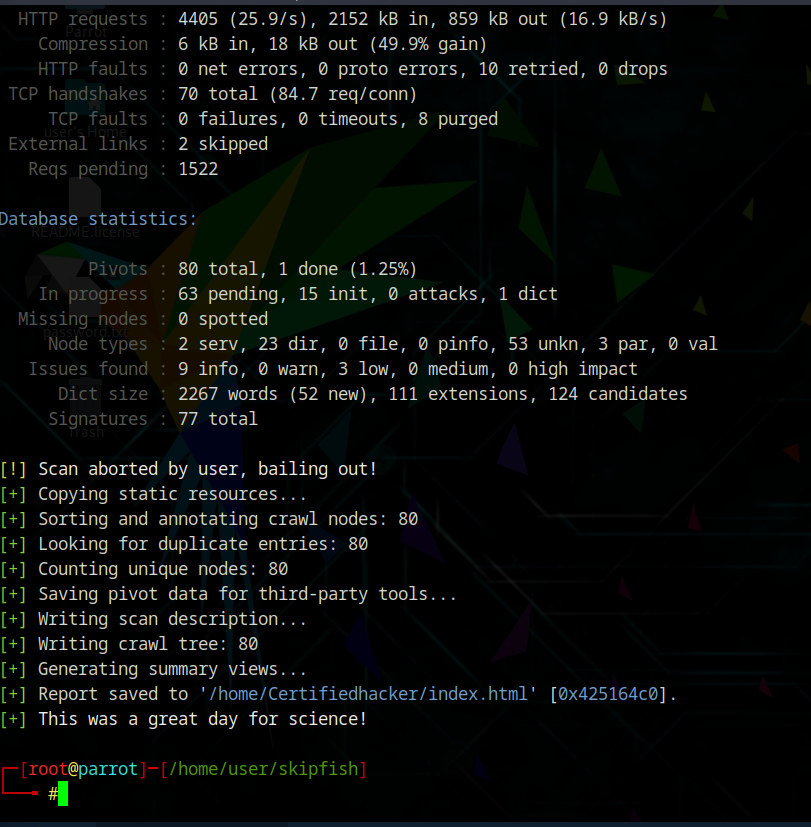


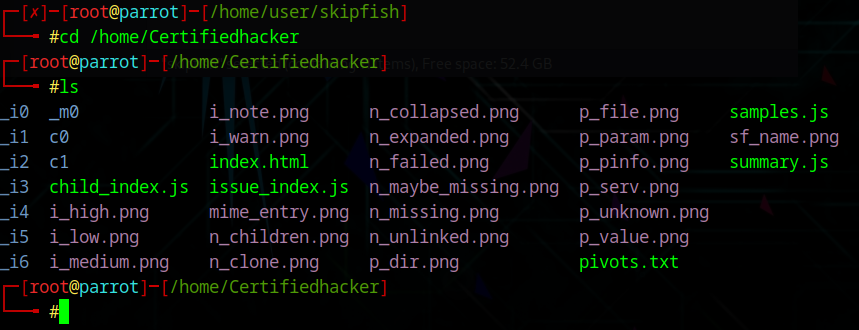


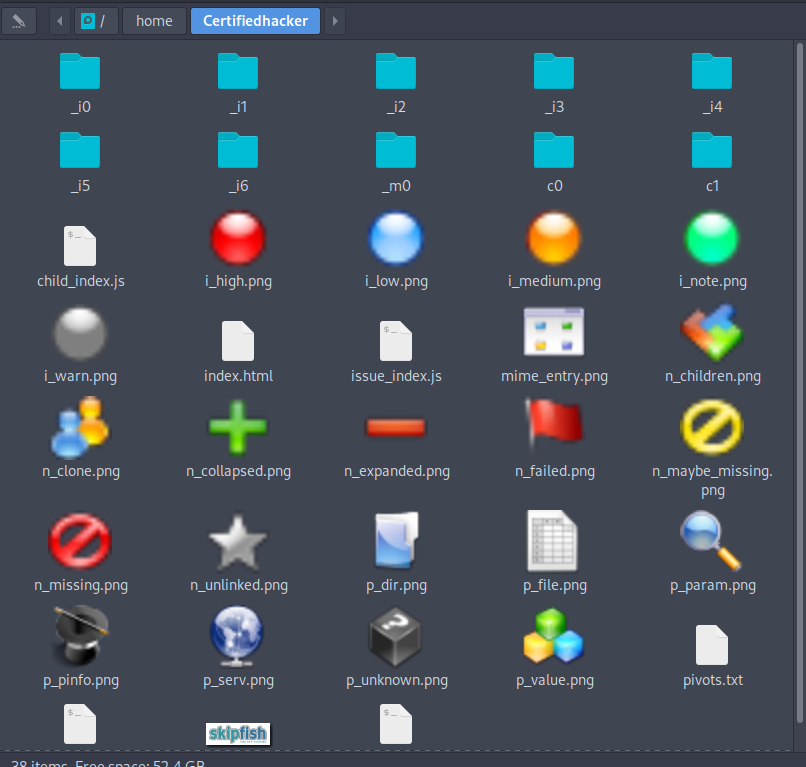


**Performing web server reconnaissance using Skipfish:**

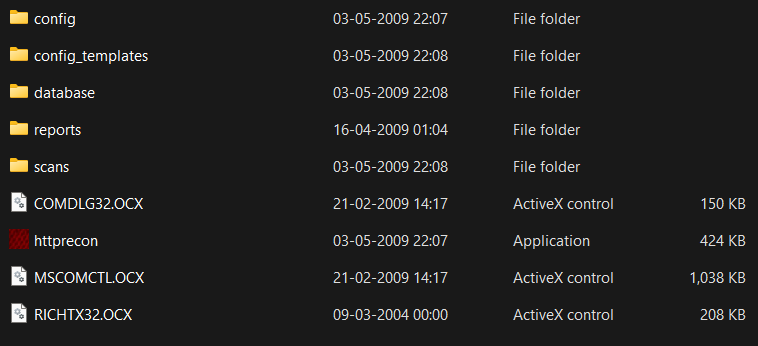




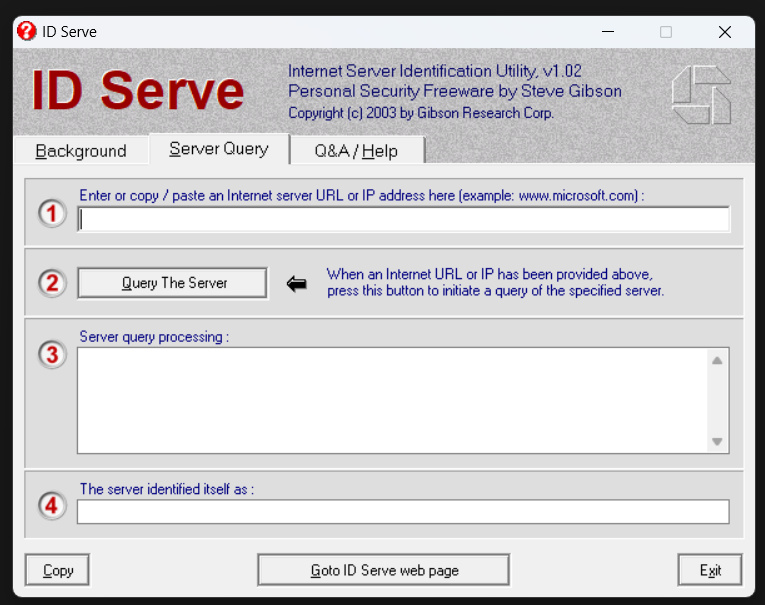


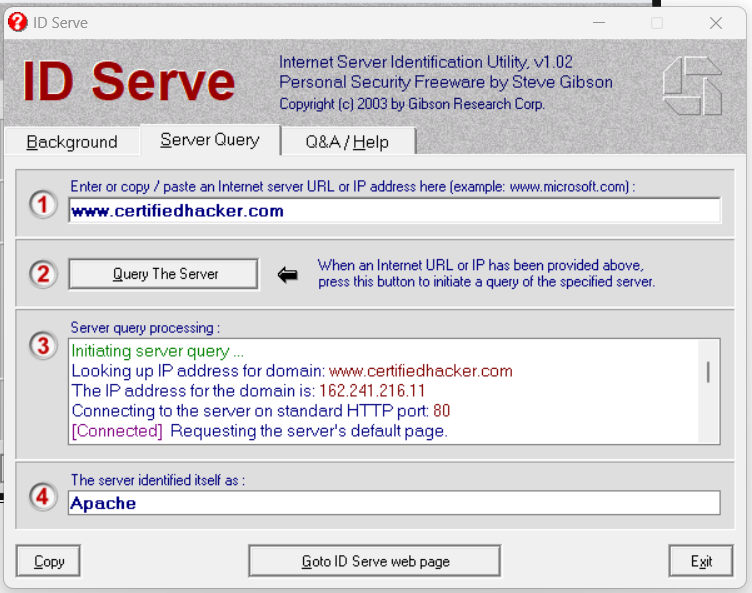


**Footprinting Webserver using Httprecon Tool:**

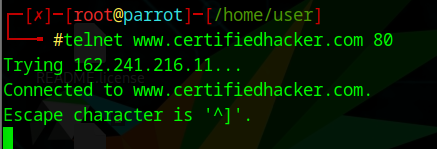
****

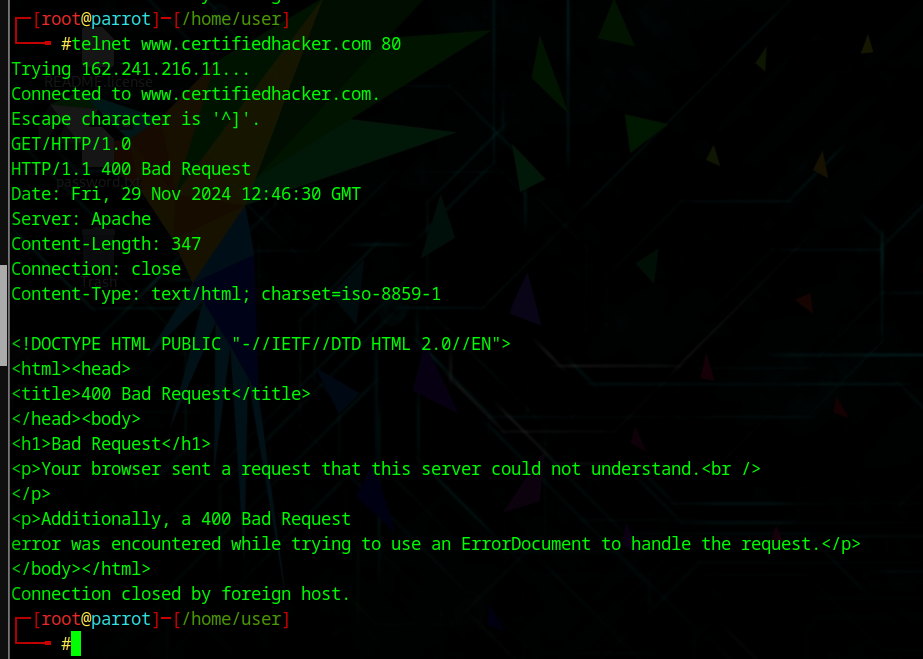
**How to use a security Investigation Tool via ID Serve:**

****

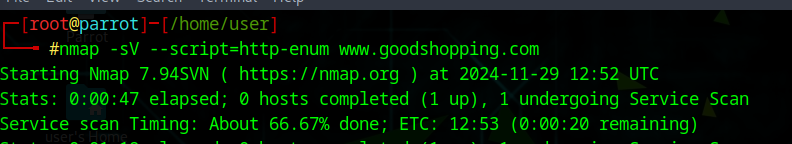
****

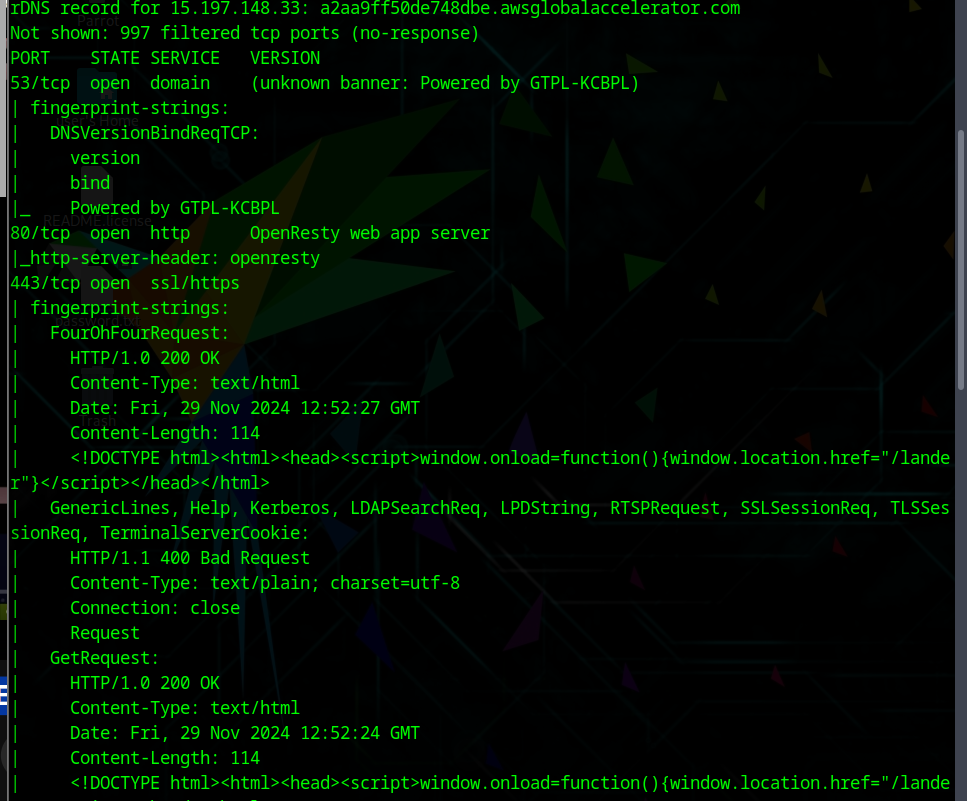
**Footprint a Web Server using Netcat and Telnet:**

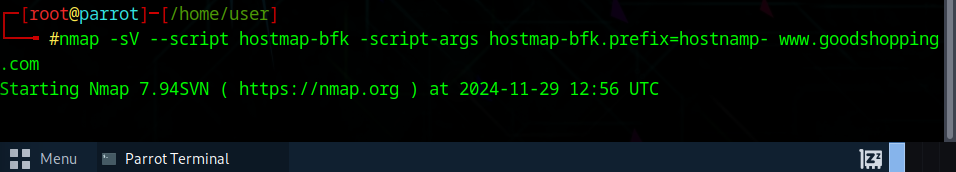


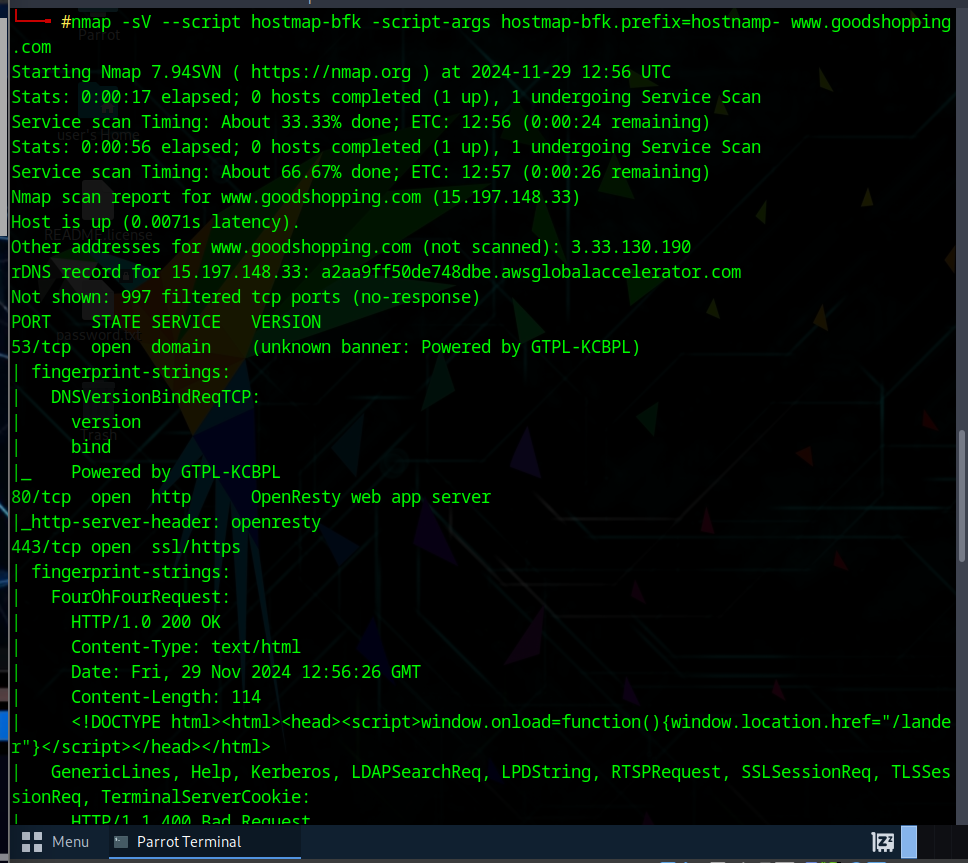


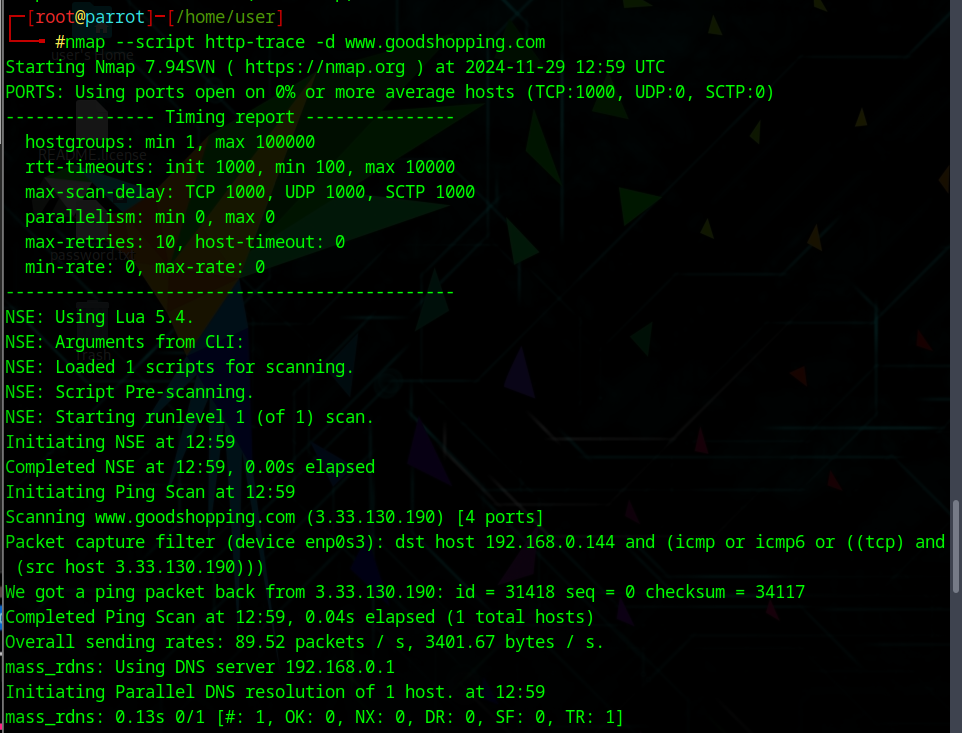
**Enumerate Web Server Information Using Nmap Scripting Engine (NSE):**

****

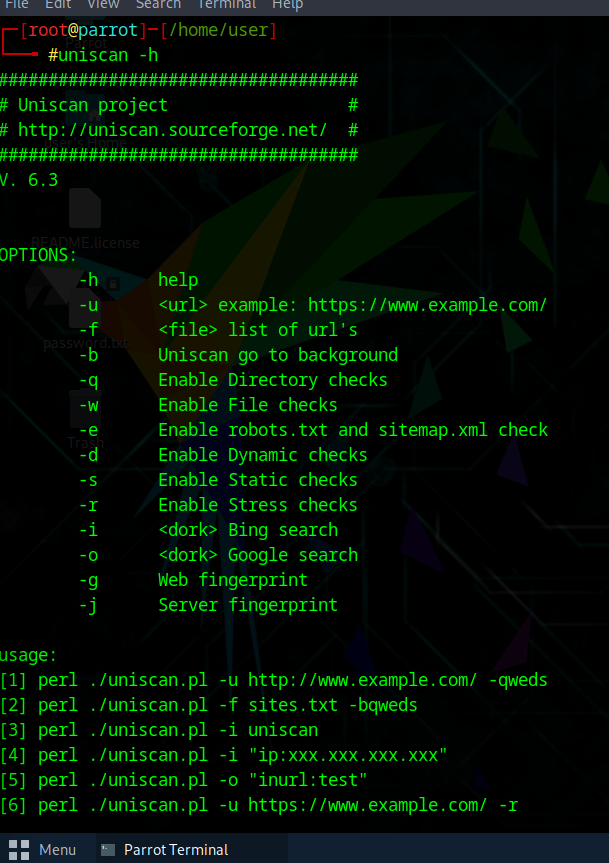
****

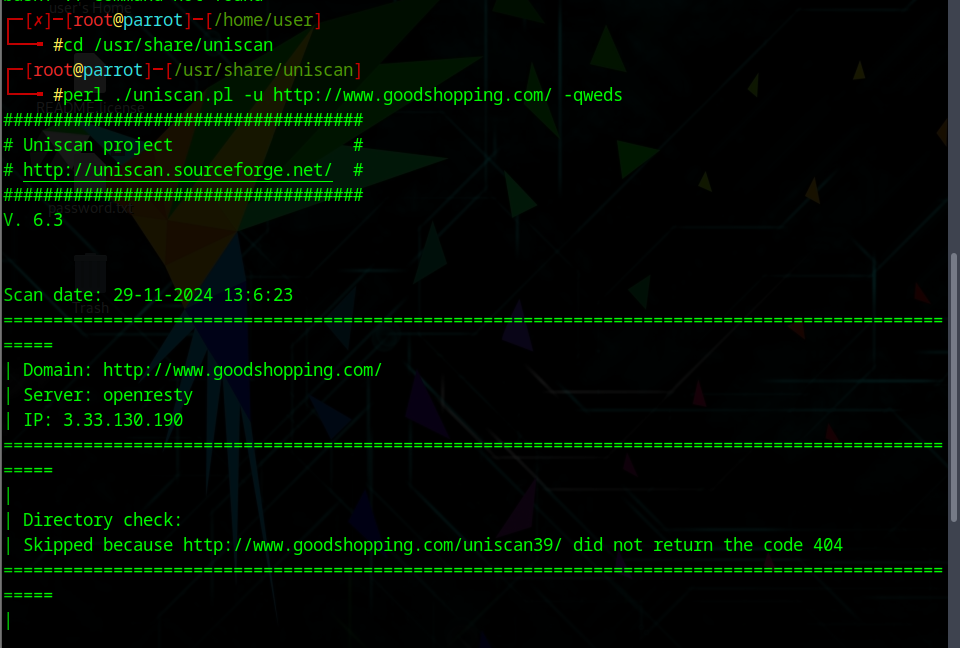


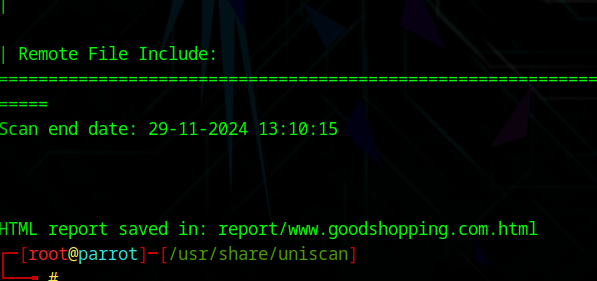




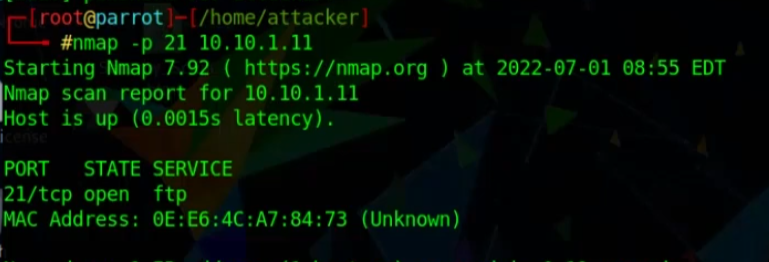
**Fingerprinting with Uniscan:**

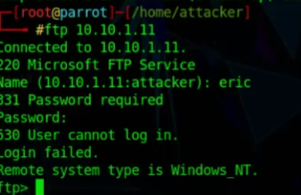


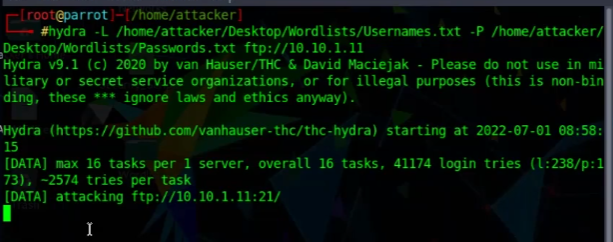




**Crack FTP Credentials Using a Dictionary Attack:**

****

****

****