**Foundational Projects: 1. Cyber Security**

### **Project 2: Conducting a Vulnerability Assessment Using Free Tools**

### **Introduction**

With the increasing number of cyber threats, vulnerability assessments are critical to identifying and mitigating potential weaknesses in systems or applications. This project guides you through conducting a basic vulnerability assessment on a local network or web application using **Nikto**, a beginner-friendly scanning tool.

The primary goal is to gain practical experience in uncovering vulnerabilities, understanding their risks, and implementing measures to secure digital infrastructure. This process is essential for developing foundational cybersecurity skills in a real-world context.

**Objective:** Perform a basic vulnerability assessment of a local web application.

**Tasks:**

1. **Install Vulnerability Scanners**: Install tools like Nikto on your local machine.

2. **Scan for Vulnerabilities**: Run scans on your local network or a web application.

3. **Analyze Results**: Review the scan results and identify potential vulnerabilities.

4. **Mitigate Vulnerabilities**: Implement mitigation measures for the identified vulnerabilities.

**Tool:** Nikto

### **Project Overview**

Set up and secure a local Apache web server, configure HTTPS, and conduct a vulnerability assessment using Nikto.

#### **Apache Installation and Configuration**

* **Apache Version**: 2.4.58
* Install Apache via XAMPP on a Windows 10 Enterprise (64-bit) machine.
* Verify Apache is running by accessing:
  + http://localhost/: Displays the Apache welcome page.
  + https://localhost/: Ensures HTTPS is enabled and functional with a self-signed SSL certificate.
* Configure SSL with:
  + **SSL Certificate**: C:\xampp\apache\conf\ssl.crt\server.crt
  + **Key File**: C:\xampp\apache\conf\ssl.key\server.key
* Update httpd-ssl.conf to set the ServerName as localhost:443.

#### **2. Firewall Configuration**

* Configure Windows Defender Firewall to allow traffic on:
  + Port 80 for HTTP.
  + Port 443 for HTTPS.
* Add custom rules to restrict unauthorized access:
  + Allow Apache HTTP Server through private and public networks.
  + Add inbound rules for specific ports.

#### **3. Generate SSL/TLS Certificates**

Use OpenSSL to generate self-signed certificates for securing the server:  
bash  
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout C:\xampp\apache\conf\ssl.key\server.key -out C:\xampp\apache\conf\ssl.crt\server.crt

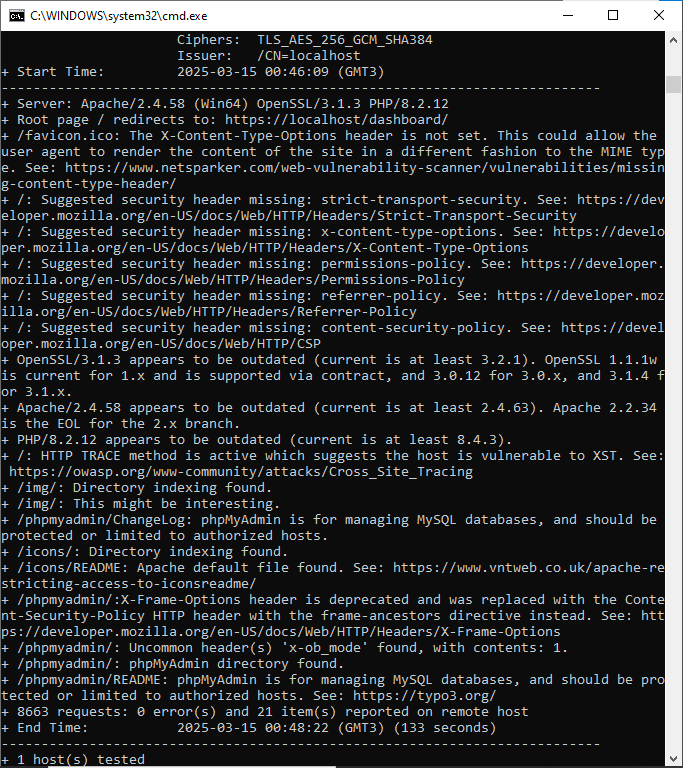
#### **4. Nikto Installation and Testing**

Clone the Nikto repository from GitHub:  
bash  
git clone https://github.com/sullo/nikto.git

* Navigate to the program directory and verify the presence of nikto.pl.

Perform a vulnerability scan on the local server using Nikto:  
bash  
perl nikto.pl -h <https://localhost>

the diagram below shows a successful vulnerability scan



#### **5. Initial Vulnerability Assessment Results**

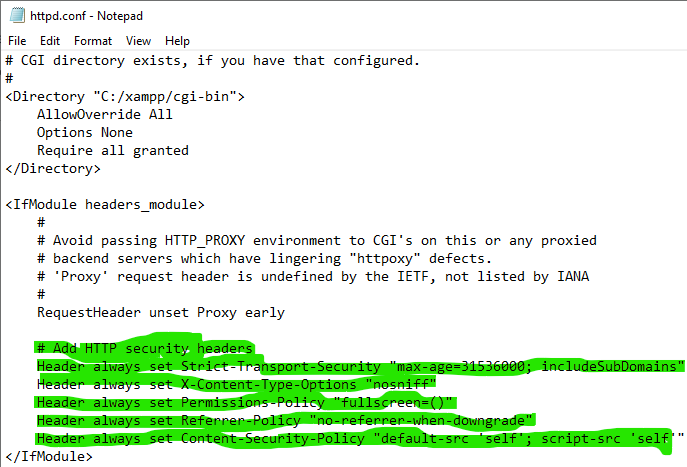
* Nikto identifies the following issues on https://localhost:

1. **Missing HTTP security headers.** Headers provide essential protection against attacks like clickjacking, MIME sniffing, and others**.** They include
   * + Strict-Transport-Security
     + X-Content-Type-Options
     + Permissions-Policy
     + Referrer-Policy
     + Content-Security-Policy
2. Outdated software:
   * + OpenSSL (3.1.3)
     + Apache (2.4.58)
     + PHP (8.2.12)
3. Directory indexing enabled in /img/ and /icons/.
4. HTTP TRACE method is active, making it vulnerable to Cross-Site Tracing (XST).
5. Exposed phpMyAdmin /phpmyadmin/ChangeLog.

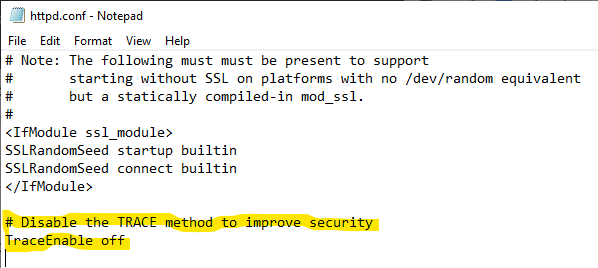
### **Next Steps**

Apply the recommended configuration updates:

1. Add missing HTTP security headers.add to your Apache configuration file (httpd.config or .htaccess): *Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains" Header always set X-Content-Type-Options "nosniff"Header always set Permissions-Policy "fullscreen=()" Header always set Referrer-Policy "no-referrer-when-downgrade" Header always set Content-Security-Policy "default-src 'self'; script-src 'self'"*

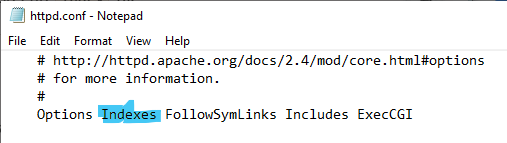


1. Disable the HTTP TRACE method. Add the following line to the configuration file: *TraceEnable off*  Once that’s done, restart the Apache server to apply the changes: *sudo systemctl restart apache2*

**

1. Restrict directory indexing. This step helps prevent unauthorized users from viewing the contents of your directories. Disable directory indexing, modify the Apache configuration or .htaccess file and add: *Options -Indexes*  Restart the Apache server to apply the update: *sudo systemctl restart apache2*

As shown in the screenshotbelow, remove the word *‘indexes’* to prevent Apache from showing the contents of directories if there isn’t an index file



1. Update Apache, PHP, and OpenSSL to their latest versions. Keeping these components up to date is crucial for maintaining security andperformance.

Use the following commands to update each software: *sudo apt update && sudo apt install --only-upgrade apache2*

**Update PHP**: **Download the Precompiled Binary:**

* Visit the PHP Windows [Downloads page.](https://windows.php.net/download/)
* Under the version you're using (e.g., PHP 8.4.5), select the **Thread Safe** version (important for most Apache setups) or **Non Thread Safe** for CLI-based use.

### **2. Replace the Files:**

* Extract the downloaded precompiled binary to your PHP installation directory (e.g., C:\php).

### **3. Restart and Verify:**

* Restart any services (like Apache) that rely on PHP.
* Run the following in Command Prompt to check: *php -v*

**Update OpenSSL**: **Download the Latest Version**:

* Visit the official OpenSSL website and [download the latest version](https://www.supportyourtech.com/articles/how-to-update-openssl-on-windows-10-a-step-by-step-guide/) for Windows.
* Choose the correct installer for your system (32-bit or 64-bit).

**Install the New Version**:

* Run the downloaded installer and follow the on-screen instructions.
* Ensure you select the appropriate options for your system.

**Verify the Update**: Open Command Prompt and type: *openssl version*

1. Re-scan the server with Nikto to confirm that vulnerabilities have been mitigated. *nikto -h https://localhost*