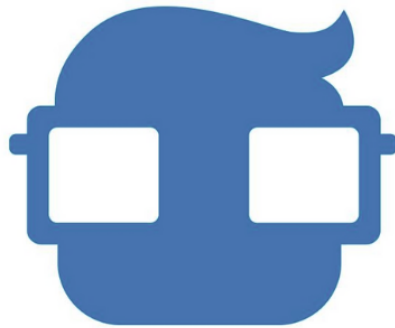


**4GEEKS ACADEMY  
BOOTCAMP DE CIBERSEGURIDAD**



# **INFORME DE VULNERABILIDADES**

**HACIENDO USO DE NMAP**

**NOMBRE DEL ESTUDIANTE: JORGE TERAN  
PROYECTO: ESCANEAR PUERTOS CON NMAP**

# Metodología

## Herramienta Utilizada - Nmap

Es una herramienta de código abierto para la exploración de redes y auditoría de seguridad, que permite identificar hosts, servicios, y vulnerabilidades en una red. Con Nmap llevaremos a cabo escaneos en una máquina objetivo (en este caso, una máquina con Debian) desde una máquina con Kali Linux.

### Paso 1: Escaneo básico con Nmap

Para comenzar el análisis, se ejecutó un escaneo básico contra la IP de la máquina objetivo utilizando el siguiente comando:

```
kali@kali:~$ nmap 10.0.2.5
```

Este comando permite detectar puertos abiertos utilizando el escaneo por defecto de Nmap, que analiza los **1.000 puertos TCP más comunes**.

A continuación, se muestra la salida obtenida:

```
(kali@kali)-[~]  
$ nmap 10.0.2.5  
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-27 07:06 EDT  
Nmap scan report for 10.0.2.5  
Host is up (0.0056s latency).  
Not shown: 998 closed tcp ports (reset)  
PORT      STATE SERVICE  
80/tcp    open  http  
443/tcp   open  https  
MAC Address: 08:00:27:D1:65:C7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)  
  
Nmap done: 1 IP address (1 host up) scanned in 0.69 seconds
```

#### Anexo 1.0

Se puede observar en el Anexo 1.0 que el Fabricante: **PCS Systemtechnik / Oracle VirtualBox virtual NIC**, Dirección MAC: **08:00:27:D1:65:C7**, esto indica que estás escaneando una máquina virtual montada en VirtualBox, asimismo dos puertos abiertos el **80 (HTTP)** y **443 (HTTPS)**. Al estar expuestos, estos puertos pueden ser analizados más a fondo para detectar posibles vulnerabilidades.

### Paso 2: Enumerar Puertos

Se ejecutó el siguiente comando (**-sV**) permite detectar la versión del servicio que está operando en cada puerto:

```
kali@kali:~$ nmap -sV 10.0.2.5
```

A continuación, se muestra la salida obtenida:

```
(kali@kali)-[~]
└─$ nmap -sV 10.0.2.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-27 07:32 EDT
Nmap scan report for 10.0.2.5
Host is up (0.0047s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE  VERSION
80/tcp    open  http     Apache httpd 2.4.62
443/tcp    open  ssl/http Apache httpd 2.4.62 ((Debian))
MAC Address: 08:00:27:D1:65:C7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: debian.debian

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.34 seconds
```

#### Anexo 1.1: Resultado de -sV mostrando Apache

Se observa en el anexo 1.1 que el servicio web corre sobre **Apache 2.4.62**, tanto en **HTTP** como en **HTTPS**. El servicio de **HTTPS (443)** está configurado como un alias de Apache con SSL (**ssl/http**), lo cual sugiere que probablemente tiene certificado auto firmado.

La información extraída es útil para investigar posibles **vulnerabilidades conocidas (CVEs)** en esa versión específica de Apache.

### Paso 3: Escaneo Detallado y Búsqueda de Vulnerabilidades

Se ejecutó el siguiente comando (**--script=vuln**) ejecuta scripts de detección de vulnerabilidades que Nmap tiene incorporados:

```
kali@kali:~$ nmap -Sv --script=vuln 10.0.2.5
```

A continuación, se muestra la salida obtenida:

```
(kali@kali)-[~]
└─$ nmap -sV --script=vuln 10.0.2.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-27 07:49 EDT
Nmap scan report for 10.0.2.5
Host is up (0.0029s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE  VERSION
80/tcp    open  http     Apache httpd 2.4.62
|_ http-server-header: Apache/2.4.62 (Debian)
|_ http-dombased-xss: Couldn't find any DOM based XSS.
|_ http-csrf: Couldn't find any CSRF vulnerabilities.
vulners:
  cpe:/a:apache:http_server:2.4.62:
    95499236-C9FE-56A6-9D7D-E943A24B633A 10.0 https://vulners.com/githubexploit/954992
36-C9FE-56A6-9D7D-E943A24B633A *EXPLOIT*
    2C119FFA-ECE0-5E14-A4A4-354A2C38071A 10.0 https://vulners.com/githubexploit/2C119F
FA-ECE0-5E14-A4A4-354A2C38071A *EXPLOIT*
    A5425A79-9D81-513A-9CC5-549D6321897C 9.8 https://vulners.com/githubexploit/A5425A
79-9D81-513A-9CC5-549D6321897C *EXPLOIT*
  CVE-2025-23048 9.1 https://vulners.com/cve/CVE-2025-23048
  CVE-2025-53020 7.5 https://vulners.com/cve/CVE-2025-53020
  CVE-2025-49630 7.5 https://vulners.com/cve/CVE-2025-49630
  CVE-2024-47252 7.5 https://vulners.com/cve/CVE-2024-47252
  CVE-2024-43394 7.5 https://vulners.com/cve/CVE-2024-43394
  CVE-2024-43204 7.5 https://vulners.com/cve/CVE-2024-43204
  CVE-2024-42516 7.5 https://vulners.com/cve/CVE-2024-42516
  CVE-2025-49812 7.4 https://vulners.com/cve/CVE-2025-49812
  CVE-2025-54090 6.3 https://vulners.com/cve/CVE-2025-54090
http-slowloris-check:
  VULNERABLE:
    Slowloris DOS attack
    State: LIKELY VULNERABLE
    IDS: CVE:CVE-2007-6750
    Slowloris tries to keep many connections to the target web server open and hold
    them open as long as possible. It accomplishes this by opening connections to
    the target web server and sending a partial request. By doing so, it starves
    the http server's resources causing Denial Of Service.
    Disclosure date: 2009-09-17
    References:
      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
      http://ha.ckers.org/slowloris/
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.
```

```
443/tcp    open  ssl/http Apache httpd 2.4.62 ((Debian))
vulners:
  cpe:/a:apache:http_server:2.4.62:
    95499236-C9FE-56A6-9D7D-E943A24B633A 10.0 https://vulners.com/githubexploit/954992
36-C9FE-56A6-9D7D-E943A24B633A *EXPLOIT*
    2C119FFA-ECE0-5E14-A4A4-354A2C38071A 10.0 https://vulners.com/githubexploit/2C119F
FA-ECE0-5E14-A4A4-354A2C38071A *EXPLOIT*
    A5425A79-9D81-513A-9CC5-549D6321897C 9.8 https://vulners.com/githubexploit/A5425A
79-9D81-513A-9CC5-549D6321897C *EXPLOIT*
  CVE-2025-23048 9.1 https://vulners.com/cve/CVE-2025-23048
  CVE-2025-53020 7.5 https://vulners.com/cve/CVE-2025-53020
  CVE-2025-49630 7.5 https://vulners.com/cve/CVE-2025-49630
  CVE-2024-47252 7.5 https://vulners.com/cve/CVE-2024-47252
  CVE-2024-43394 7.5 https://vulners.com/cve/CVE-2024-43394
  CVE-2024-43204 7.5 https://vulners.com/cve/CVE-2024-43204
  CVE-2024-42516 7.5 https://vulners.com/cve/CVE-2024-42516
  CVE-2025-49812 7.4 https://vulners.com/cve/CVE-2025-49812
  CVE-2025-54090 6.3 https://vulners.com/cve/CVE-2025-54090
|_ http-server-header: Apache/2.4.62 (Debian)
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_ http-csrf: Couldn't find any CSRF vulnerabilities.
|_ http-dombased-xss: Couldn't find any DOM based XSS.
|_ http-enum:
  /wordpress/: Blog
  /wordpress/wp-login.php: Wordpress login page.
MAC Address: 08:00:27:D1:65:C7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: debian.debian

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 75.80 seconds
```

#### Anexo 1.2: Resultado de -sV --script=vuln

En el **Anexo 1.2** se observa un escaneo con **nmap** usando el script **vuln** sobre la IP **10.0.2.5**, el cual identificó múltiples vulnerabilidades asociadas con el servidor web **Apache HTTPD 2.4.6** en los puertos **80** y **443**.

A continuación, te presento tabla de las vulnerabilidades encontradas, debido a que ambos puertos (**80** y **443**) están sirviendo el mismo servicio (**Apache HTTP Server 2.4.57**), comparten las **mismas vulnerabilidades**:

Puerto	Servicio	Versión	CVE ID	Enlace	Riesgo
			CVE-2025-23048	<a href="https://vulners.com/cve/CVE-2025-23048">https://vulners.com/cve/CVE-2025-23048</a>	9.1 (Alta)
			CVE-2025-53020	<a href="https://vulners.com/cve/CVE-2025-53020">https://vulners.com/cve/CVE-2025-53020</a>	7.5 (Alta)
			CVE-2025-49630	<a href="https://vulners.com/cve/CVE-2025-49630">https://vulners.com/cve/CVE-2025-49630</a>	7.5 (Alta)
80/tcp	Apache HTTP	2.4.57	CVE-2024-47252	<a href="https://vulners.com/cve/CVE-2024-47252">https://vulners.com/cve/CVE-2024-47252</a>	7.5 (Alta)
			CVE-2024-43394	<a href="https://vulners.com/cve/CVE-2024-43394">https://vulners.com/cve/CVE-2024-43394</a>	7.5 (Alta)
			CVE-2024-43204	<a href="https://vulners.com/cve/CVE-2024-43204">https://vulners.com/cve/CVE-2024-43204</a>	7.5 (Alta)
			CVE-2024-42516	<a href="https://vulners.com/cve/CVE-2024-42516">https://vulners.com/cve/CVE-2024-42516</a>	7.5 (Alta)
			CVE-2025-49812	<a href="https://vulners.com/cve/CVE-2025-49812">https://vulners.com/cve/CVE-2025-49812</a>	7.4 (Alta)
			CVE-2025-23048	<a href="https://vulners.com/cve/CVE-2025-23048">https://vulners.com/cve/CVE-2025-23048</a>	9.1 (Alta)
			CVE-2025-53020	<a href="https://vulners.com/cve/CVE-2025-53020">https://vulners.com/cve/CVE-2025-53020</a>	7.5 (Alta)
			CVE-2025-49630	<a href="https://vulners.com/cve/CVE-2025-49630">https://vulners.com/cve/CVE-2025-49630</a>	7.5 (Alta)
443/tcp	Apache HTTPS	2.4.57	CVE-2024-47252	<a href="https://vulners.com/cve/CVE-2024-47252">https://vulners.com/cve/CVE-2024-47252</a>	7.5 (Alta)
			CVE-2024-43394	<a href="https://vulners.com/cve/CVE-2024-43394">https://vulners.com/cve/CVE-2024-43394</a>	7.5 (Alta)
			CVE-2024-43204	<a href="https://vulners.com/cve/CVE-2024-43204">https://vulners.com/cve/CVE-2024-43204</a>	7.5 (Alta)
			CVE-2024-42516	<a href="https://vulners.com/cve/CVE-2024-42516">https://vulners.com/cve/CVE-2024-42516</a>	7.5 (Alta)
			CVE-2025-49812	<a href="https://vulners.com/cve/CVE-2025-49812">https://vulners.com/cve/CVE-2025-49812</a>	7.4 (Alta)