	Informe de análisis de vulnerabilidades, explotación y resultados del reto MONKEY.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	21/04/2024	22/04/2024	1.0	MQ-HM-MONKEY	RESTRINGIDO

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

N3- MQ-HM-MONKEY

Generado por:

JUC4ZU

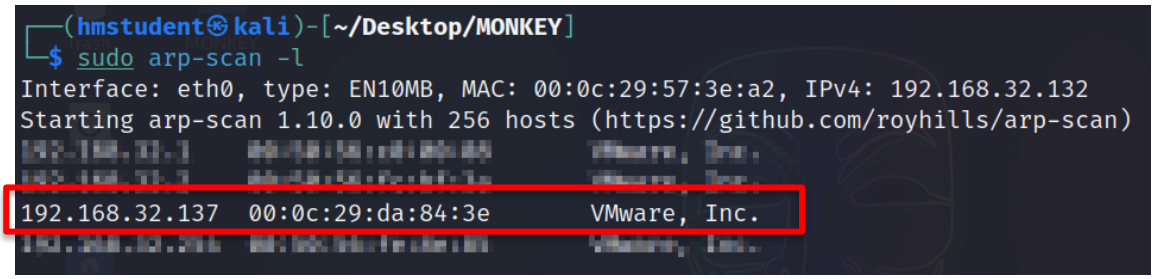
Estudiante de Hacker Mentor

Fecha de creación:
21.04.2024

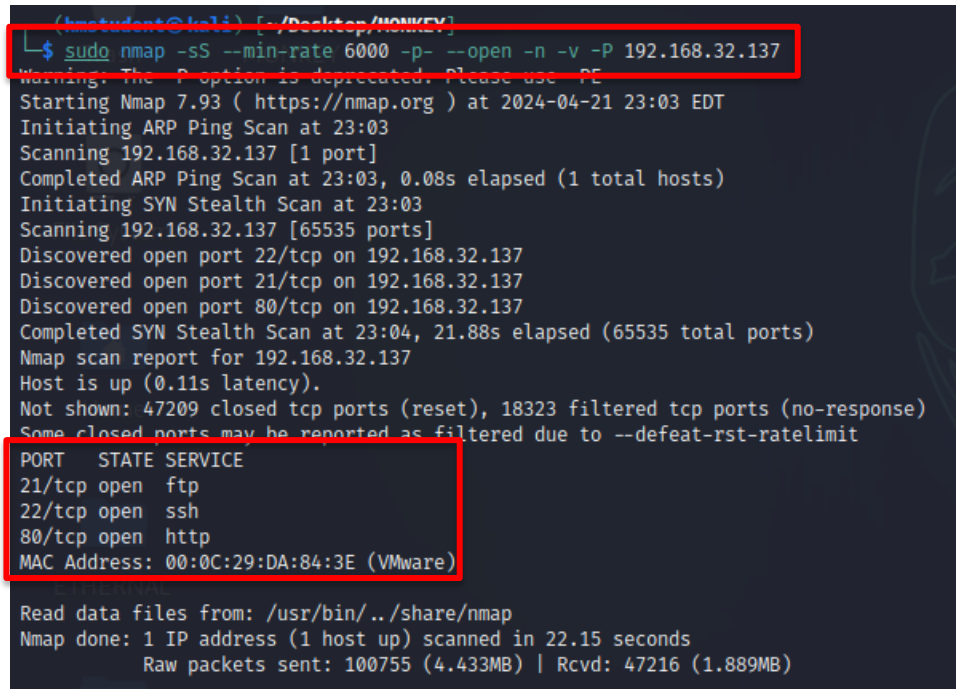
Índice

1. Reconocimiento

Ubicamos los datos del equipo victima:



Haremos la verificación de puertos abiertos:



Tenemos los siguientes puertos abiertos, 21, 22, 80. Ya con esto podemos generar un documento con las vulnerabilidades del equipo.

Ports

The 997 ports scanned but not shown below are in state: **closed**

- 997 ports replied with: **reset**

Port		State (toggle closed [0] | filtered [0])	Service	Reason	Product	Version	Extra info
21	tcp	open	ftp	syn-ack	vsftpd	3.0.3	
	ftp-anon	Anonymous FTP login allowed (FTP code 230) -rw-r--r-- 1 1000 1000 791 May 15 2022 notas.txt					
	ftp-syst	STAT: FTP server status: Connected to ::ffff:192.168.32.132 Logged in as ftp TYPE: ASCII No session bandwidth limit Session timeout in seconds is 300 Control connection is plain text Data connections will be plain text At session startup, client count was 2 vsFTPd 3.0.3 - secure, fast, stable End of status					
22	tcp	open	ssh	syn-ack	OpenSSH	7.9p1 Debian 10+deb10u2	protocol 2.0
	ssh-hostkey	2048 c744588690fde4de5b0dbf078d055dd7 (RSA) 256 78ec470f0f53aaa6054884809476a623 (ECDSA) 256 999c3911dd3553a0291120c7f8bf71a4 (ED25519)					
80	tcp	open	http	syn-ack	Apache httpd	2.4.38	(Debian)
	http-title	Apache2 Debian Default Page: It works					
	http-server-header	Apache/2.4.38 (Debian)					

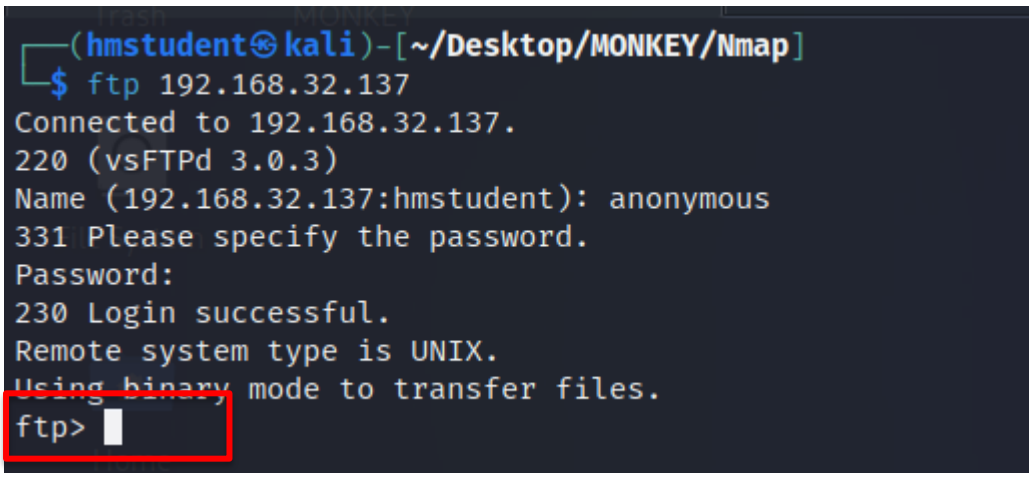
Buscando entre las vulnerabilidades encontramos que el FTP permite conectar con un usuario "Anonymous" – Esta es una debilidad que podemos explotar. Empezaremos verificando desde este punto para recabar información.

IP, Puertos Sistema operativo

IP	192.168.32.137
Sistema Operativo	Linux 4.15 - 5.6
Puertos/Servicios	21 ftp 22 ssh 80 http

2. Análisis de vulnerabilidades/debilidades

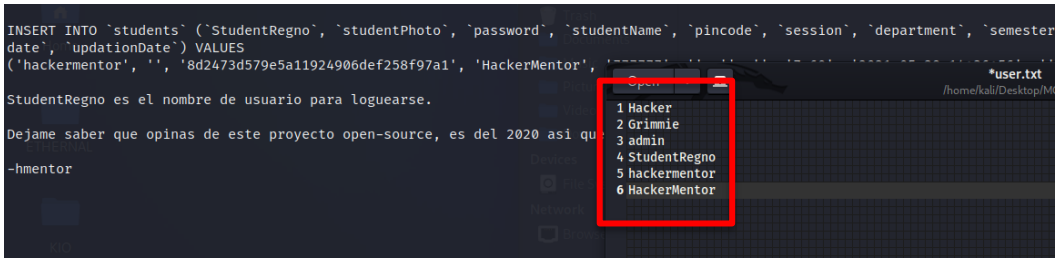
Ya teniendo el primer paso para atacar, vamos a ingresar al "**FTP**" de esta máquina:



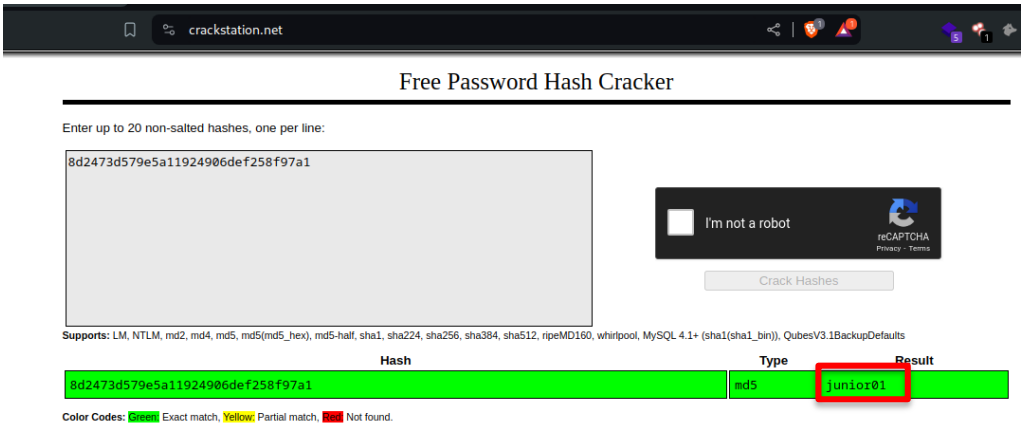
Aquí dentro podemos ubicar un documento con pistas:

```
ftp> ls
229 Entering Extended Passive Mode (|||14151|)
150 Here comes the directory listing.
-rw-r--r--    1 1000     1000          791 May 15  2022 notas.txt
226 Directory send OK.
ftp> more notas.txt
Hola Hacker !
Grimmie está probando el sitio web para la nueva academia.
Le dije que no utilice la mismo contraseña en otros servicios y que la cambie lo más pronto posible.

No pude crear un usuario a traves del panel de admin, entonces lo agregué directamente en la base de datos con el siguiente comando:

INSERT INTO `students` (`StudentRegno`, `studentPhoto`, `password`, `studentName`, `pincode`, `session`, `department`, `semester`, `cgpa`, `creation
date`, `updationDate`) VALUES
('hackermentor', '', '8d2473d579e5a11924906def258f97a1', 'HackerMentor', '777777', '', '', '', '7.60', '2021-05-29 14:36:56', '');

StudentRegno es el nombre de usuario para loguearse.

Dejame saber que opinas de este proyecto open-source, es del 2020 asi que deberia ser seguro, verdad?

-hmentor
```

De aquí podemos sacar cierta información, como usuarios, un "**hash**" de contraseña que puede ser muy valioso más adelante, así que esta información la vamos a capturar en documentos para utilizarlos más tarde.

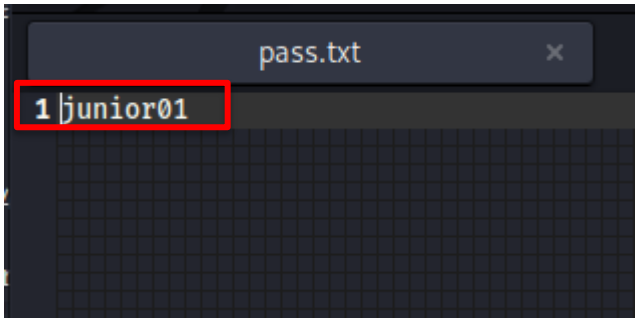


Y el hash de contraseña podemos descifrarlo desde (https://crackstation.net/)



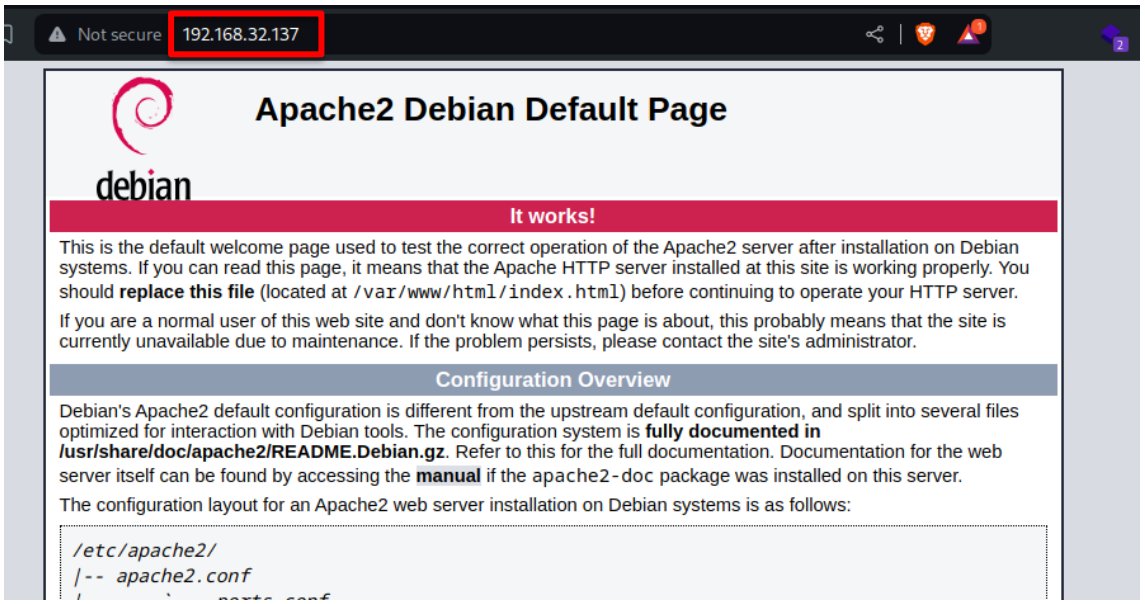
Este seria nuestro punto de partida para verificar el "**Apache**" que está en el puerto 80 y tratar de obtener credenciales para atacar el "**SSH**" del equipo víctima y de esa forma tomar acceso completo.

Guardamos la contraseña que pudimos obtener:

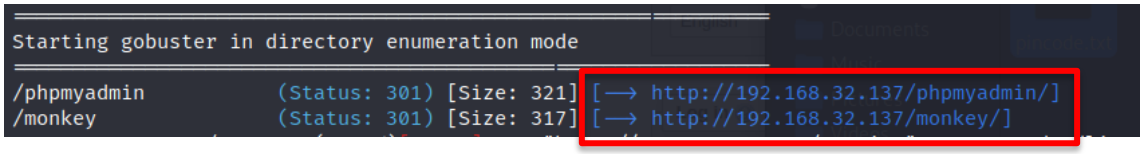


Y verificaremos la página asociada a esta computadora para descartar o intentar conectarnos con los datos anteriores.

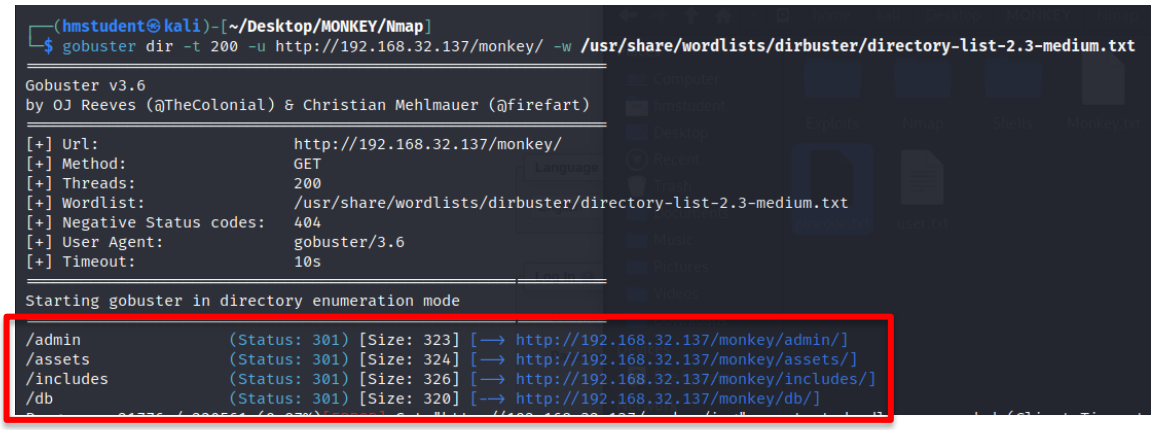
Al ingresar notamos que el apache puede estar sin configurar o en "**Default**", esta es otra falla.



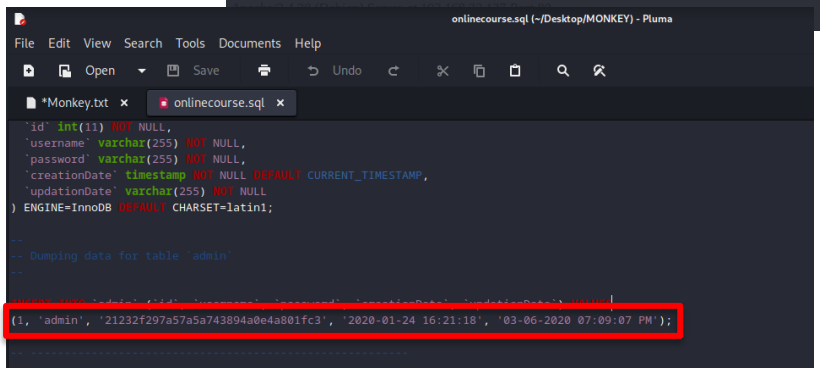
Aplicaremos un escaneo de direcciones web o "**fuzzing**" con la aplicación "**Gobuster**" (instalaremos si no lo tenemos **sudo apt install gobuster**)



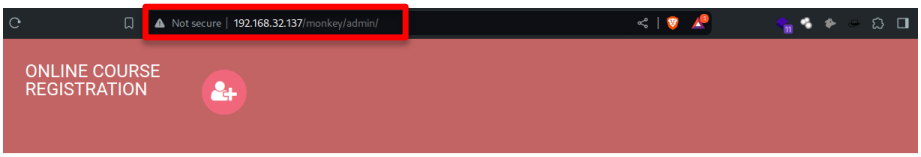
Nos brindará algunas direcciones que pueden ser importantes, como la de "**Monkey**" que puede ser escaneada nuevamente para encontrar alguna vulnerabilidad.

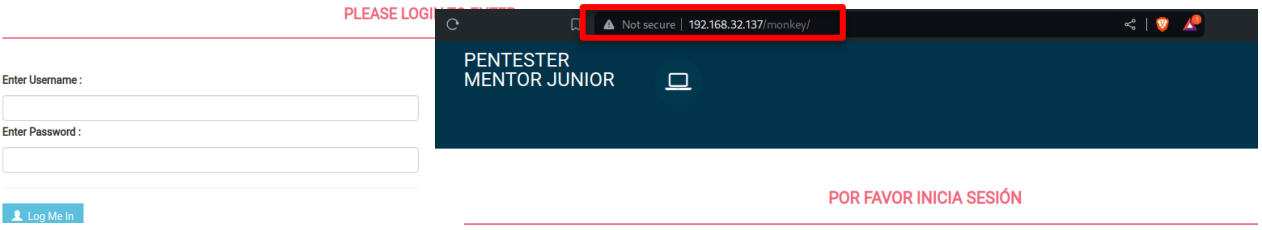


En los enlaces que obtuvimos nos pueden importar el "**admin**", o el "**db**" que pueden contener información que nos ayude.



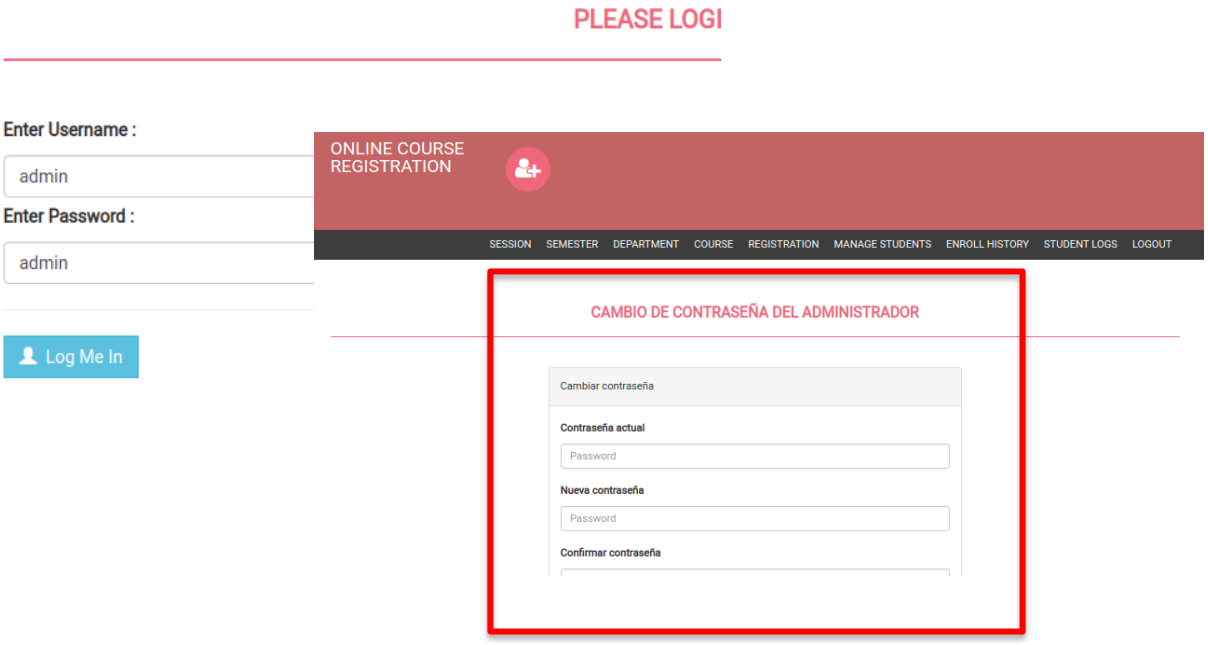
Por ejemplo, en "**db**" podemos descargar una base de datos que contiene una credencial que podríamos utilizar en una de las páginas activas en este "**server**".



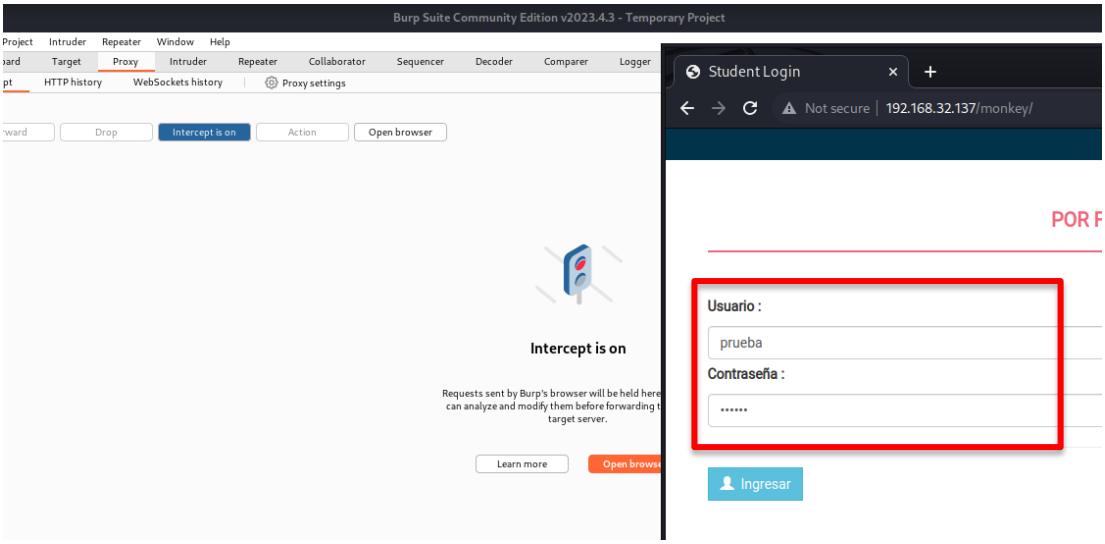


Navegando en el sitio podemos encontrar 2 páginas importantes donde intentar ingresar.

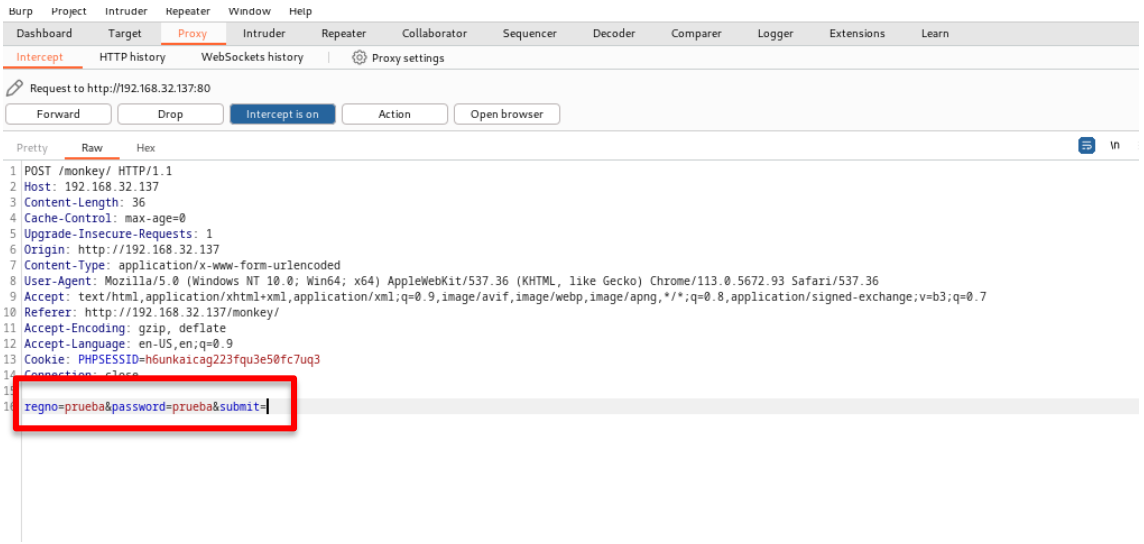
Por ejemplo, en la página "**admin**" podemos acceder fácilmente con la contraseña que haciendo la prueba con las credenciales que conseguimos, en la base de datos anterior en donde el usuario es "**admin**" y su contraseña (**21232f297a57a5a743894a0e4a801fc3**) también es "**admin**". Aunque ingresando de esta manera no tenemos demasiada libertad.



Intentaremos en la otra página que sería el inicio de sesión para estudiantes de esta página, aplicando fuerza bruta con "**burpsuite**" y la lista de contraseñas que hemos conseguido:

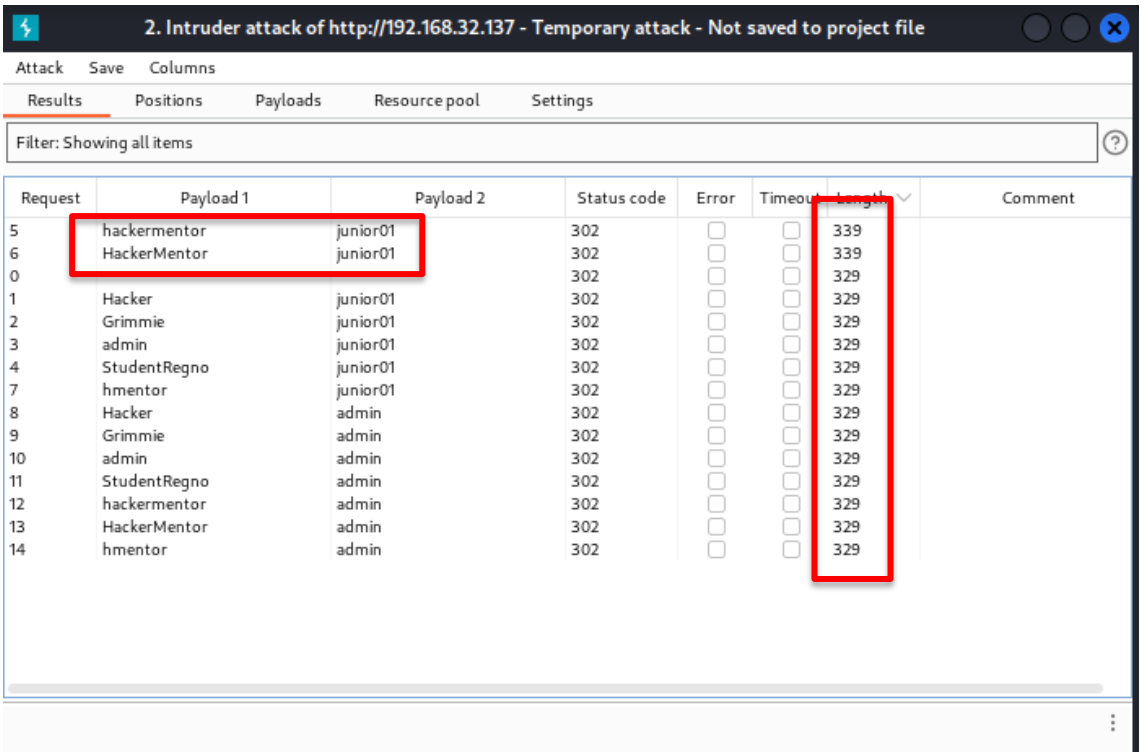


Haremos una prueba de acceso, interceptando los datos del "**login**" y aprovecharemos esta consulta de la página intentar forzar nuestro ataque.



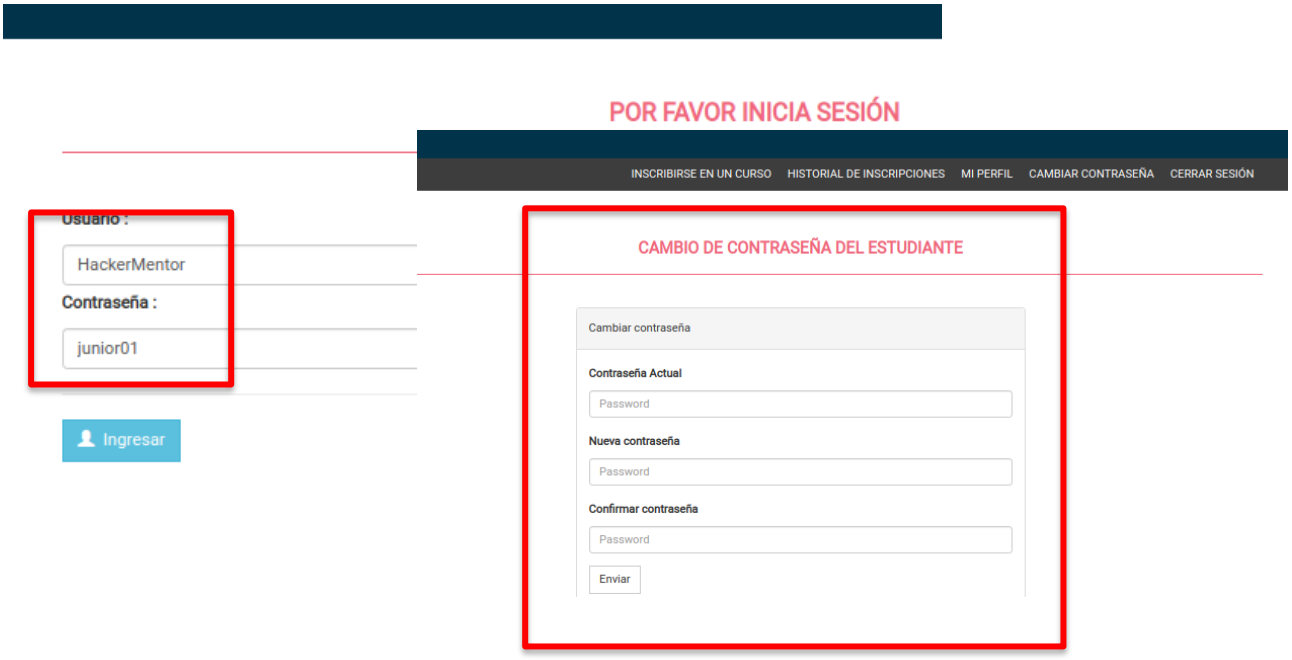
Obtendremos una respuesta de la página con nuestros datos de prueba que ingresamos, ajustaremos las variables para que apliquen la fuerza bruta, cargaremos como "**payload**" un "**Cluster Bomb**", además pondremos nuestra lista de usuarios y contraseñas almacenadas para probar si podemos acceder.

Al hacer esta prueba podemos observar que la página nos regresa diferentes longitudes de datos, en este caso, las respuestas que más pueden acercar a las credenciales correctas tienen un largo un poco mayor a las incorrectas.

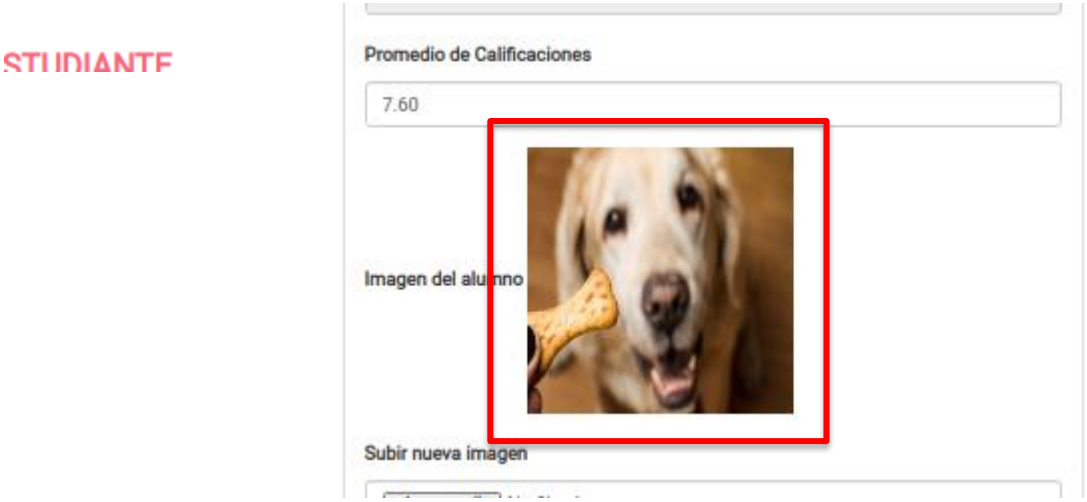


Request	Payload 1	Payload 2	Status code	Error	Timeout	Length	Comment
5	hackermentor	junior01	302			339	
6	HackerMentor	junior01	302			339	
0			302			329	
1	Hacker	junior01	302			329	
2	Grimmie	junior01	302			329	
3	admin	junior01	302			329	
4	StudentRegno	junior01	302			329	
7	hmentor	junior01	302			329	
8	Hacker	admin	302			329	
9	Grimmie	admin	302			329	
10	admin	admin	302			329	
11	StudentRegno	admin	302			329	
12	hackermentor	admin	302			329	
13	HackerMentor	admin	302			329	
14	hmentor	admin	302			329	

Sabiendo esto, encontramos que las respuestas algo más extensas contienen el usuario "**HackerMentor**" o "**hackermentor**" y ambos con la contraseña "**junior01**". Haremos la prueba con estas credenciales.



Ya obteniendo el acceso a esta página podemos realizar pruebas en los diferentes enlaces, aunque el que más nos puede llamar la atención es "**Mi Perfil**"



Ya que desde acá podemos subir archivos directamente al servidor víctima y eso nos permitirá tomar acceso.

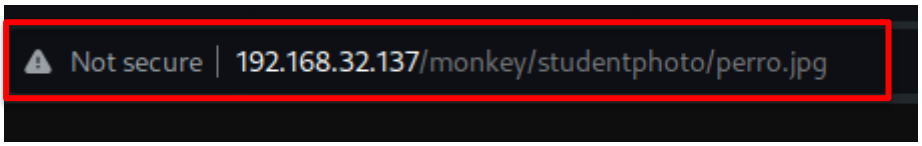
Haciendo pruebas, y con ayuda de herramientas como "**Gobuster**" o "**burpsuite**" pudimos obtener acceso a este sitio web. Y desde aquí completaremos la explotación ya que considero que apenas tengamos la propiedad de cambiar algo dentro de la máquina es cuando de verdad la vulneramos.

Puerto	**Vulnerabilidad**
21	vsftpd 3.0.3 – Usuario y Contraseña "Anonymous"
22	OpenSSH | 7.91p1 Debian 10+deb10u2 – Permite acceder con usuarios del equipo que cuentan con contraseñas débiles – Inclusive root
80	Apache httpd 2.4.38 – Cuenta con una versión de apache con carencias para defenderse de SQL Injection o subida de archivos sin restricción. Configuración por defecto que puede ser fácil de atacar.

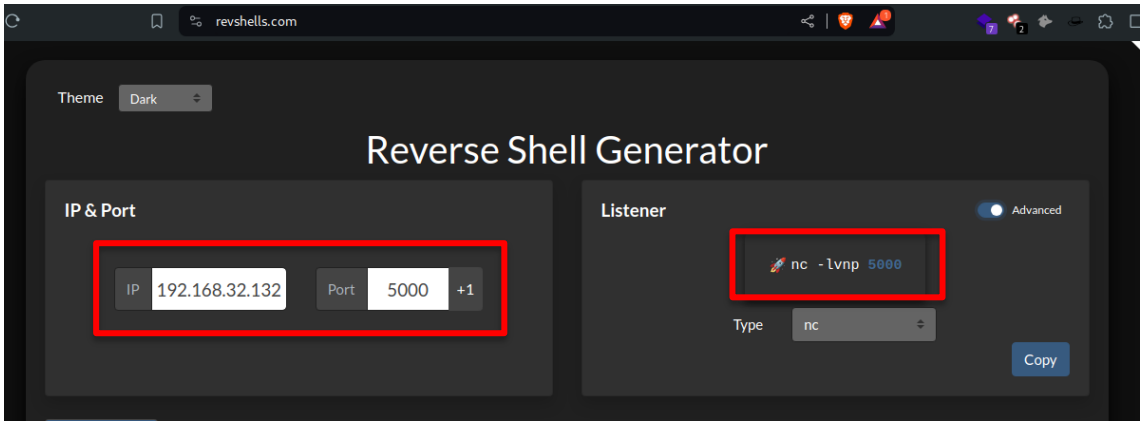
3. Explotación

Manual

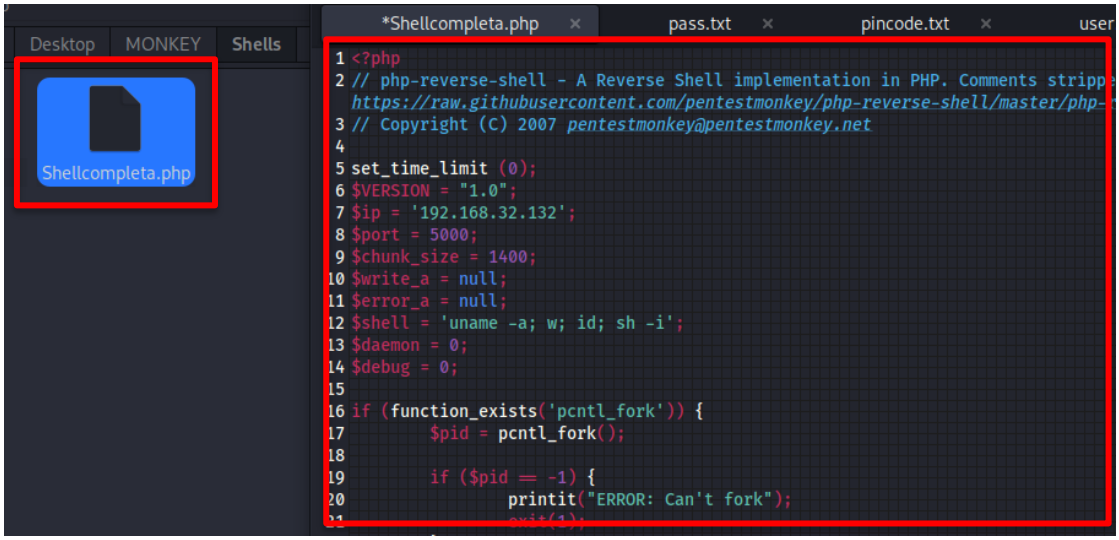
Ya con la propiedad de poder subir archivos al servidor, podemos intentar agregar un "**Shell reverse**" dentro de está máquina para ver si esta forma la controlamos.



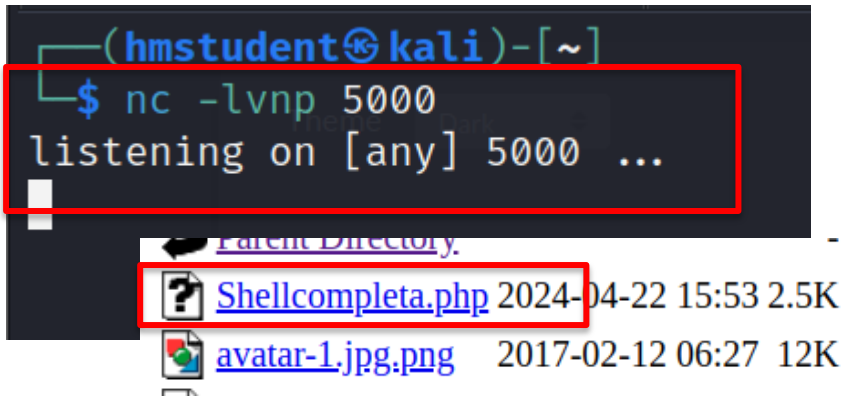
Nos dirigiremos a la página Reverse (https://www.revshells.com/) desde aquí podemos generar un código para que el equipo víctima sea quien se conecte a nosotros.



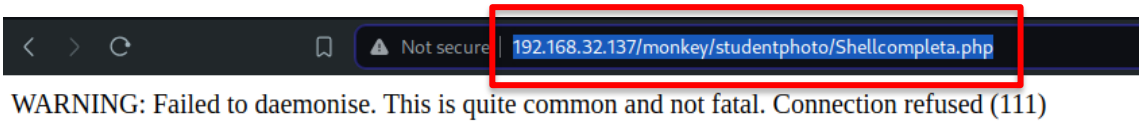
Configuramos los datos según nuestra computadora, y debemos activar un servidor de escucha para conectarnos. Procederemos generando un archivo "**php**" donde vamos a guardar el "**Shell completo**" generado de la página que acabamos de visitar, y el cual subiremos a la web de estudiantes.



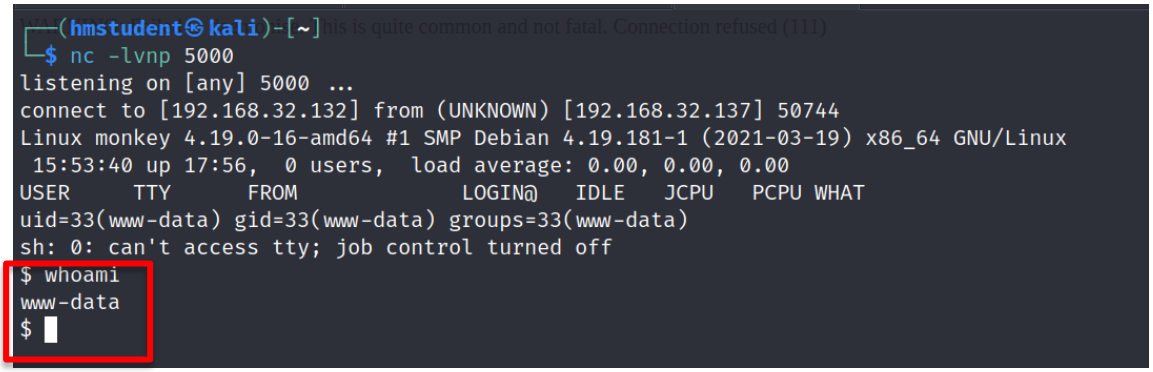
Activaremos nuestro servidor de escucha, subiremos el archivo a la página web.



Al cargar la dirección web de donde esta nuestra "**Shell**" reversa, se nos hará una conexión con nuestro computador:



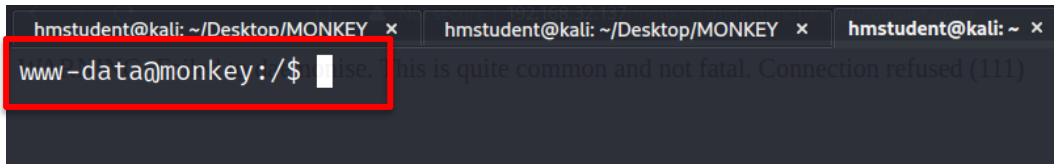
Con esto ya estaremos dentro del equipo, pero con permisos básicos del usuario que utiliza "**Apache**", "**www-data**".



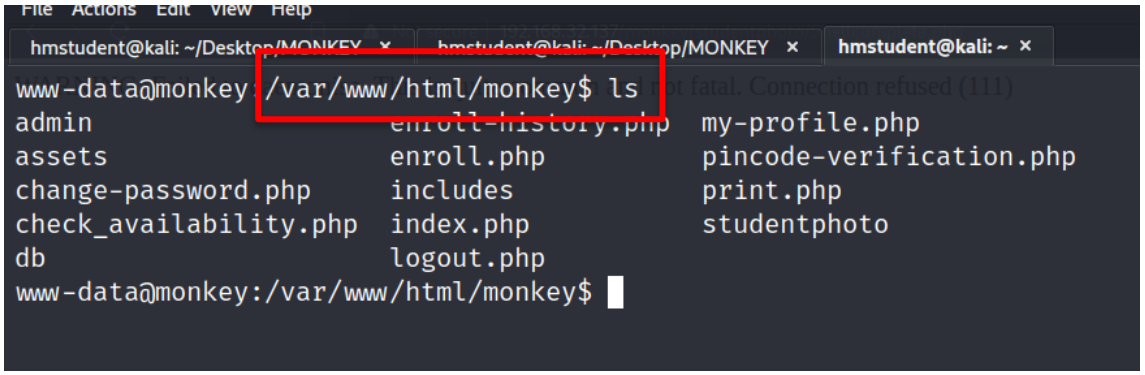
Aplicaremos los comandos para mejorar la "**Shell**" utilizada en este pasó para que nos sea más sencillo su uso y navegación.

```
script /dev/null -c bash
crtl +z
stty raw -echo; fg
reset
xterm
echo $TERM
TERM=xterm
echo $SHELL
SHELL=bash
```

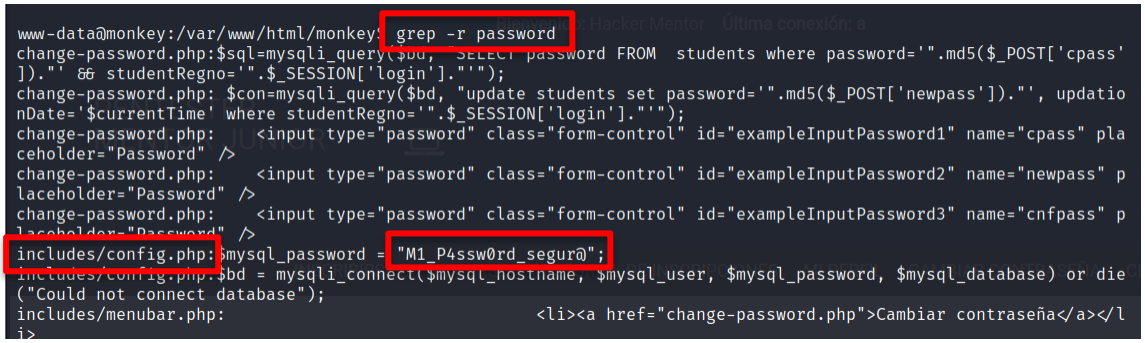
Ya con la terminal mejorada realizaremos, la búsqueda de directorios con información valiosa.



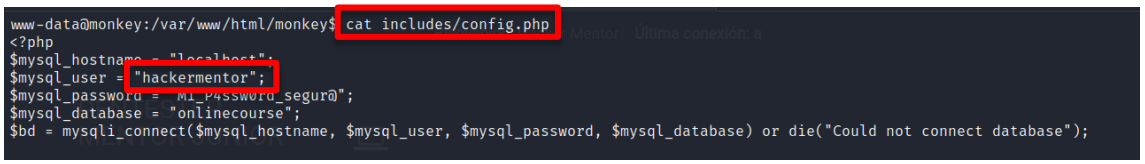
Entre toda esta información, nos podemos dirigir a una ubicación importante desde donde se guardan los archivos de configuración del "**Apache**" – "**/var/www/html/monkey**", desde aquí podemos buscar información que pueda representar un riesgo para el equipo.



Ejecutaremos el siguiente comando para buscar entre todos estos datos:
"**grep -r password**" con este comando conseguiremos alguna contraseña desprotegida contenida en todos estos datos.

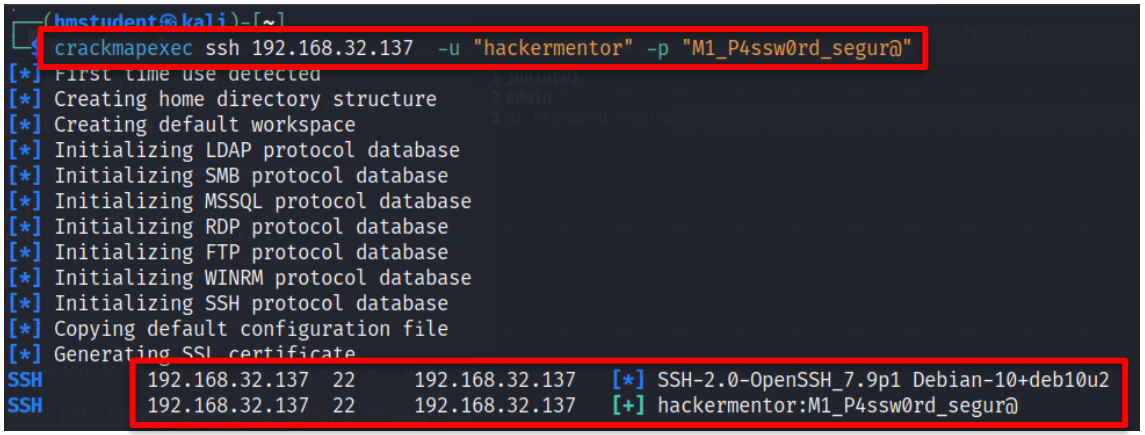


Si observamos en toda esta amalgama de letras, podemos obtener una contraseña que hace referencia a una base de datos de "**MySQL**" y una ubicación que podemos consultar para saber que usuario puede estar enlazado a la misma:

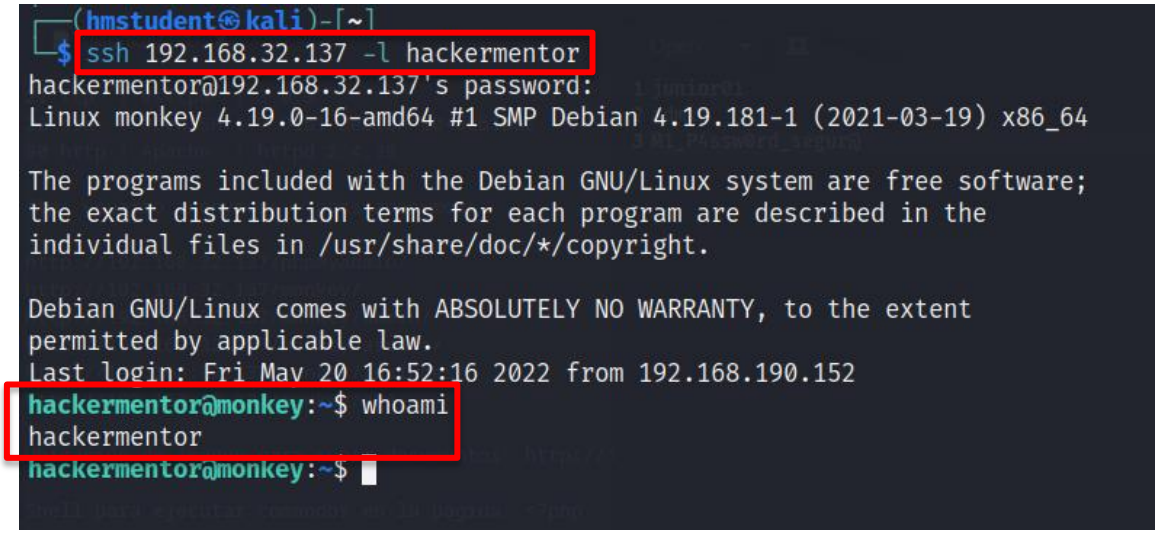


Con lo anterior visto, obtuvimos un usuario y una contraseña, haciendo memoria una de las pistas de esta máquina, indica que uno de los administradores tiene el hábito de utilizar las mismas credenciales para todo, así que intentaremos forzar la misma en el servicio de "**SSH**" así poder ingresar efectivamente.

Así que haremos una prueba con este comando para forzar las credenciales en el puerto 22 "**crackmapexec ssh 192.168.32.137 -u "hackermentor" -p "M1_P4ssw0rd_segur@""**



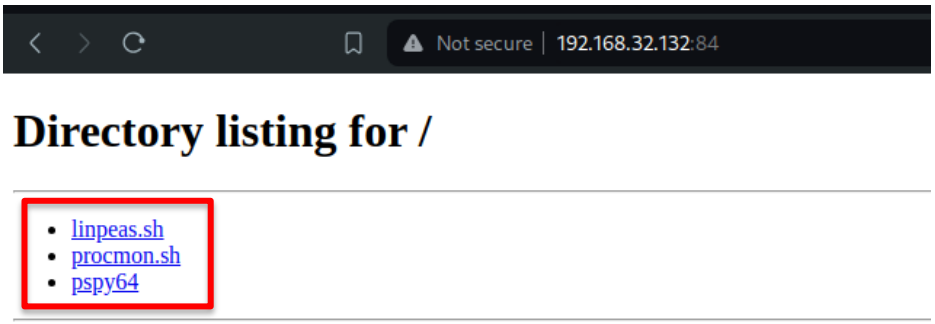
Al final de este, nos muestra que el equipo victima es accesible con estas credenciales que acabamos de probar, así que nos aprovecharemos para acceder.



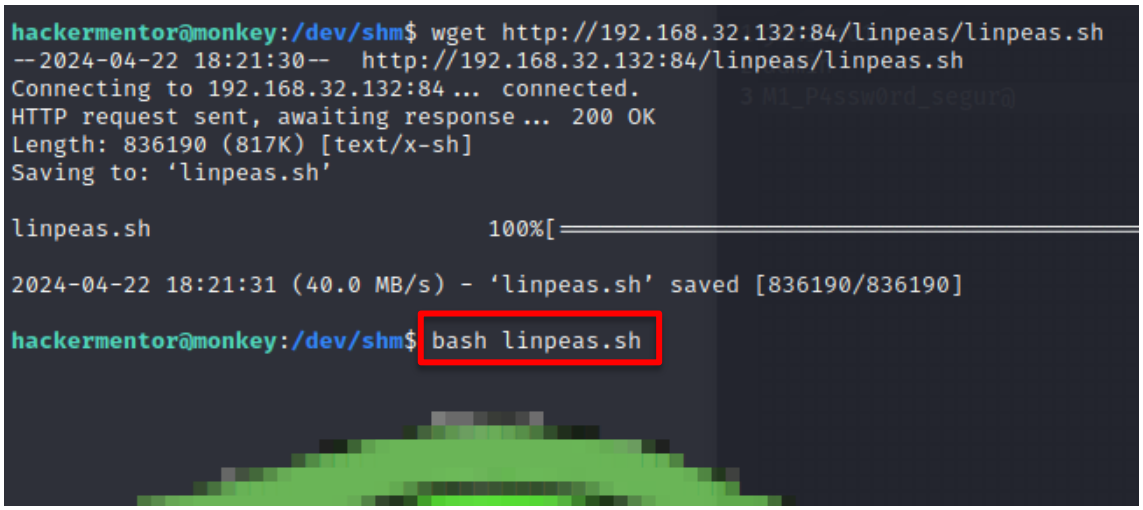
Probando el comando para acceder por "**SSH**" nos damos cuenta de que, si nos funciona y podemos tomar control del equipo, aunque sin cantar victoria porque aún no somos "root".

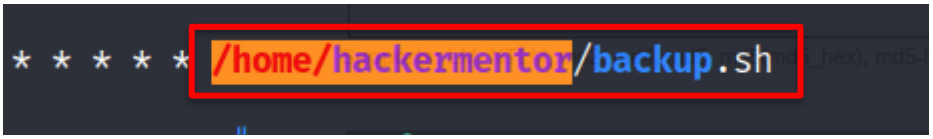
Para lo siguiente primero debemos analizar que procesos se están ejecutando en el equipo que puedan representar un núcleo de ataque. Esto lo haremos cargando aplicaciones como "**Linpeas**", "**PSPY64**" o "**Procmon**". Así que habilitaremos un servidor http en una carpeta de nuestro equipo atacante para descargar todos estos paquetes dentro de la víctima.

Con el servidor arriba, procederemos a copiar los ejecutables al equipo que estamos atacando para realizar pruebas de identificación de vectores vulnerables. Los copiaremos en la ubicación **/dev/shm** – Como medida de prevención para ser descubiertos.



Si descargamos y ejecutamos el "**Linpeas**" podremos obtener una ubicación importante donde se está ejecutando un archivo por parte del usuario "**root**"

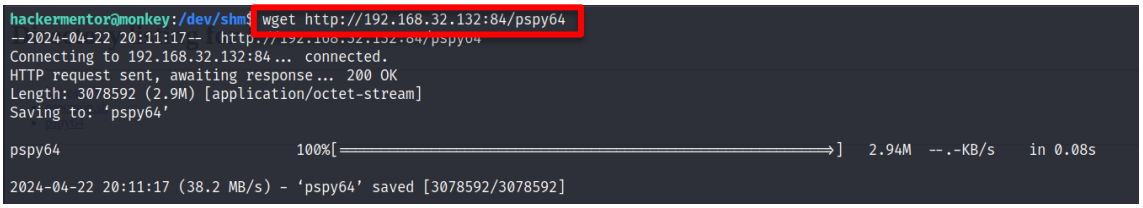




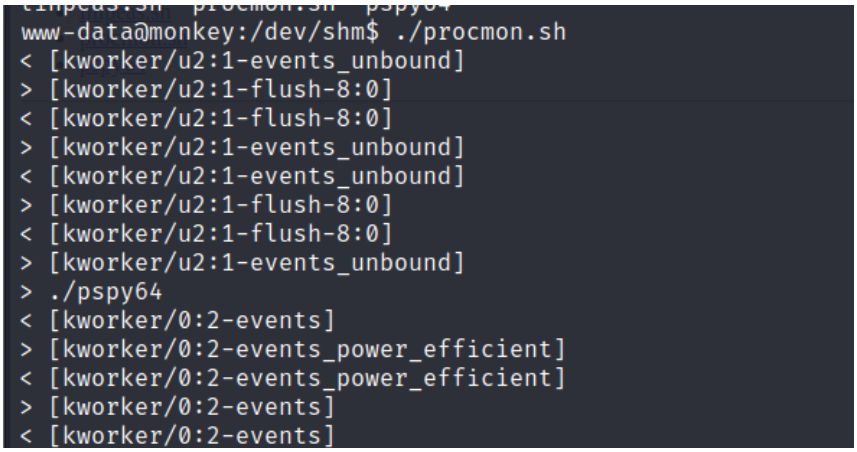
Esta dirección es importante, aunque no la tenemos completa, debemos verificar si es la misma que podemos observar desde el usuario "hackermentor".

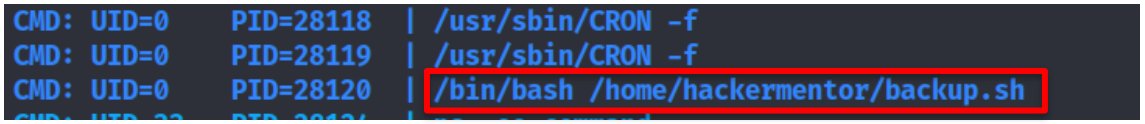
Para esto podemos hacer una ejecución alterna entre "**Procmon**" y "**PSYP64**" ya que nos podrán mostrar en pantalla todos los procesos que están corriendo dentro de la víctima.

De la misma forma, descargaremos ambos “**Scripts**” dentro de la máquina y los ejecutaremos:



Los pondremos en ejecución al mismo, desde terminales diferentes:

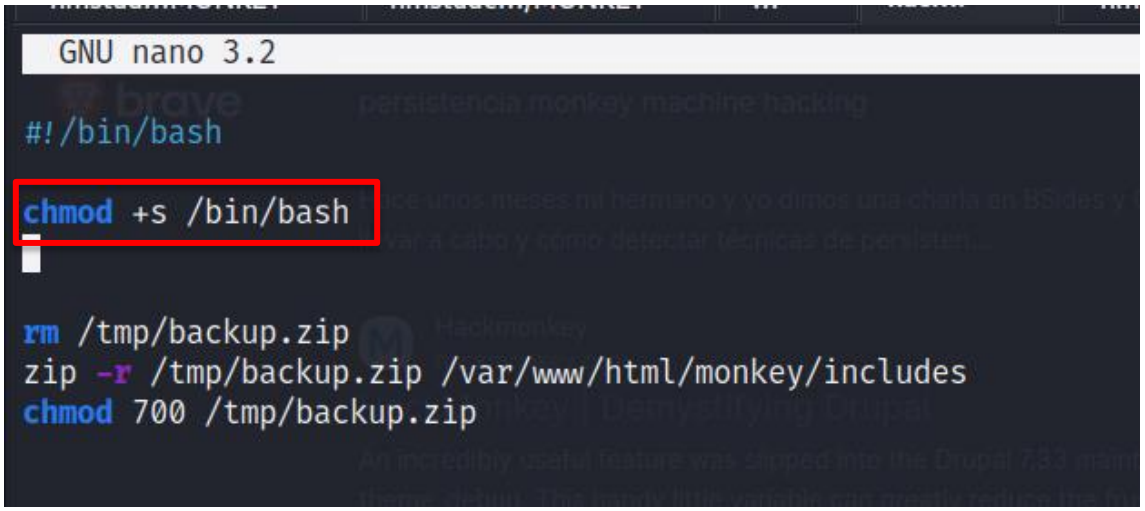




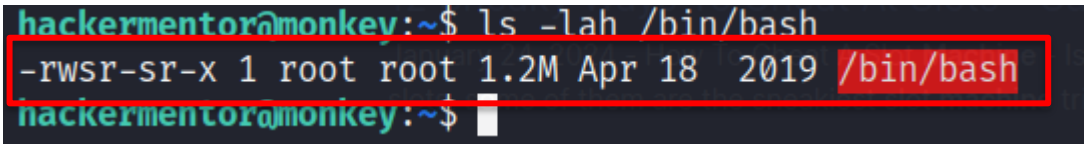
Mientras se ejecutan ambos, hay una tarea, automática que se aplica cada 60 segundos aproximadamente, esta es muy parecida a la que nos muestra el “**Linpeas**”, así que esta es nuestra pista para darnos cuenta de que ese archivo es más importante de lo que parece.

Así que sabiendo esto, podemos tratar de editar el script que se esta corriendo en bash, y trastocarlo para ejecutar lo que nos permita tomar el control total.

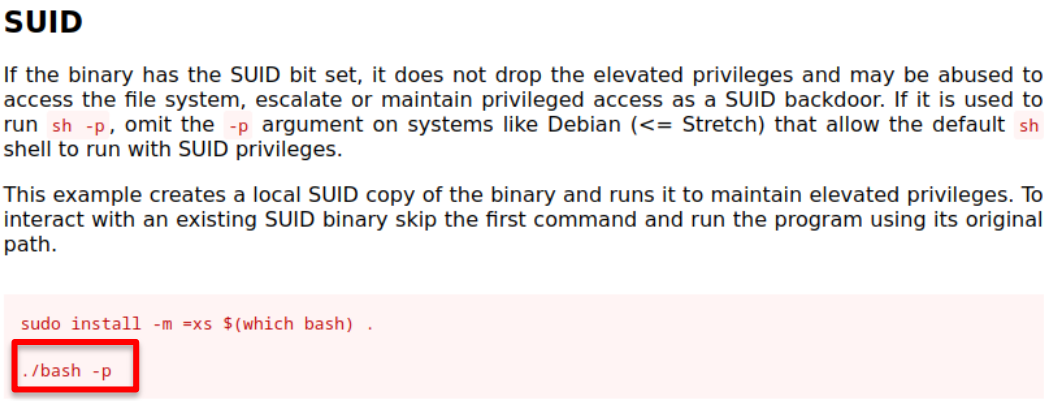
Abriremos el archivo con un editor de texto, y como el usuario “hackermentor” tiene permisos de edición podemos agregar un cambio de permiso sobre el “**/bin/bash**”, que esto va a provocar que las ejecuciones de código se realicen como “**root**”.



Aceptaremos los cambios y guardaremos el documento. Solo nos quedará esperar hasta que el proceso se vuelva a ejecutar.

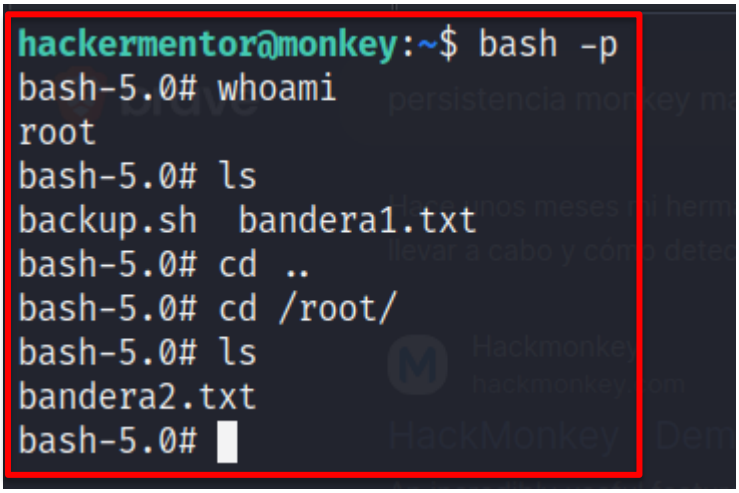


Al ejecutarse el proceso automático, el “**/bin/bash**” ya habrá sido alterado, esto nos permite ejecutar un escalado de privilegios como se muestra en esta página: (https://gtfobins.github.io/gtfobins/bash/)



Así que aplicaremos el comando directamente en la consola del equipo vulnerado.

Lo ejecutaremos como se nos indica, con "**bash -p**", esto causa que escalemos a "**root**" directamente. Y si hacemos la comprobación ya tendremos acceso a las banderas del equipo atacado.



4. Escalación de privilegios si

Se realiza por medio de escalación de SUID, esto implica que se hará ejecución de un archivo que es de uso exclusivo de root, para transformar un usuario convencional a root automáticamente.

5. Banderas

Bandera1	47ee0702e489445bae251df46bc88b73
Bandera2	d844ce556f834568a3ffe8c219d73368

6. Herramientas usadas

Gobuster	Enumeración de direcciones - Fuzzing
Crackmapexec	Comprobación de usuarios
Nmap	Enumeración de puertos y servicios
Linpeas	Ubicaciones de importancia para atacar
PSPY64	Verificación de procesos
Burpsuite	Ataque de fuerza bruta web
Procmon	Ejecución de procesos de contraste

7. EXTRA Opcional

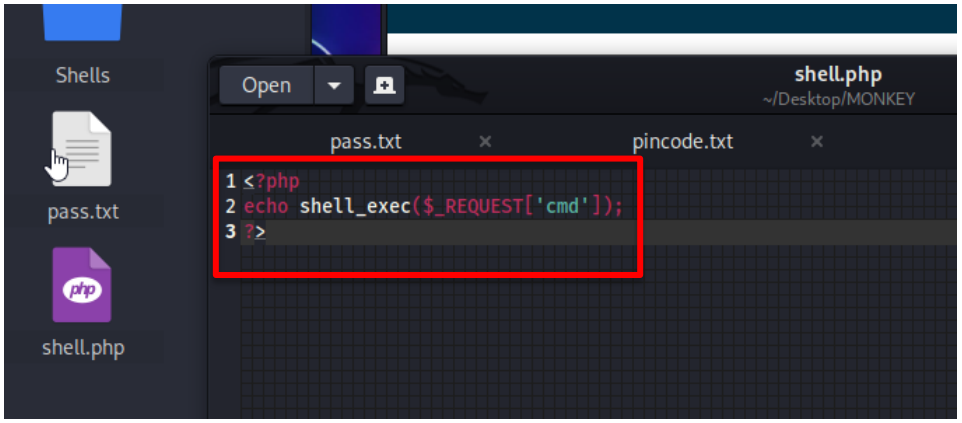
Herramientas usadas

PHP	La ejecución de código en la página web

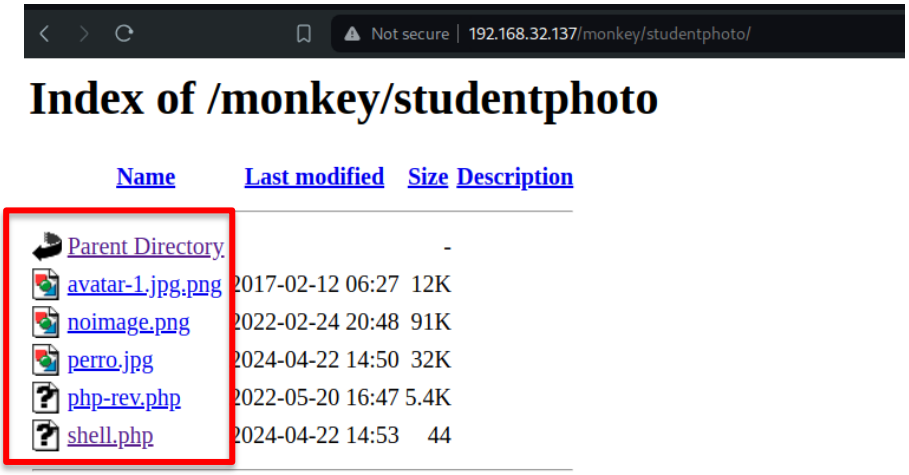
PUNTO EXTRA

Explicaré la forma en que podemos ver la bandera1 en el navegador.

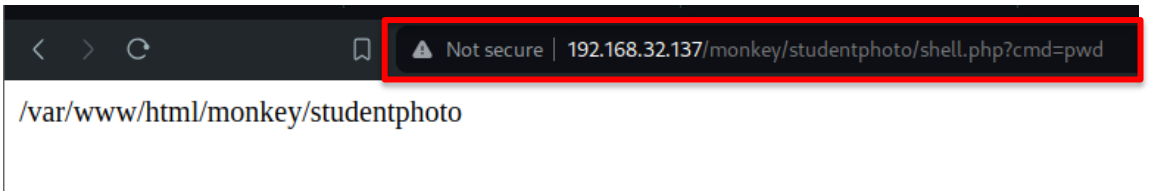
Podemos generar una shell para subir a la página con la función de subir imagen. Este código debe ir dentro de un archivo "**.php**".



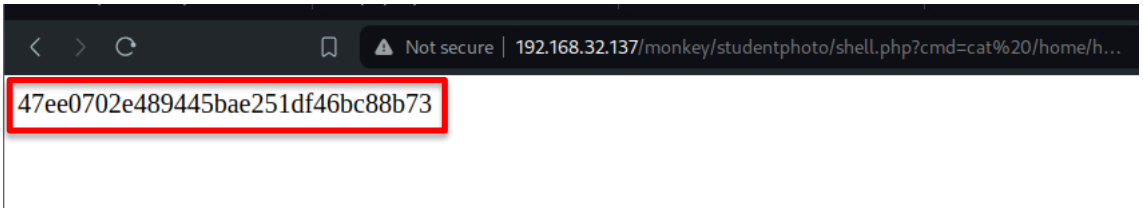
Esto se utilizará como una terminal básica en el navegador. La subiremos a la página para verlo.



Daremos clic sobre el "**Shell.php**" que subimos por acá, esto nos mostrara una dirección en la parte superior que podemos editar. Agregando un "?**cmd=pwd**" al final podemos provocar comando en la página que nos muestran información.



Ya sabiendo esto solo agregamos la dirección de la bandera, anteponiendo la palaba cat para abrirla:



Y listo con este tendremos la vista de nuestra bandera.

8. Conclusiones y Recomendaciones

1)Evitar utilizar contraseñas débiles para el FTP o SSH.
2)Configurar adecuadamente los servidores web con Apache (No dejarlos predeterminados).
3)Limitar los archivos que se puedan subir a la web que publiquemos o eliminar los que puedan representar un peligro.
4)Actualizar los Servicios que utilicemos para nuestras páginas.
5)No dejar pistas que puedan comprometernos.
6)Seguir una correcta política de creación de contraseñas en especial para los administradores.
7)Proteger los accesos por direcciones aleatorias o generadas por alguna herramienta para evitar el “Fuzzing” de la web.