🔒 Enumeration for Linux Privilege Escalation

■ Linux

g0jirasan gojirasan 2 ✎ May '17

So, you have a shell on a Linux machine. But what now? Privilege Escalation! (Unless you spawned a root shell of course! Then its time for some lateral movement I suppose). But before you can escalate your privileges, you will have to figure out how you are going to do it. So that brings us to enumeration, which is hands down the most important part of compromising a target.

So this post is not intended to be a definitive guide to Linux priv-esc or anything, but merely a simple collection of things that I have personally found helpful during enumeration. I am totally open to suggestions or any ideas. This post is also heavily inspired by g0tmi1k's amazing post, Basic Linux Privilege Escalation:

https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/ 363

I recommend bookmarking that ^

Get Your Bearings

First things first. Always get a good feel for the machine. Its always a good idea to figure out what version you're looking at:

```
cat /etc/issue
```

or

```
cat /etc/*-release
```

What is the kernel version? Are there known exploits for that version?

```
cat /proc/version
```

```
uname -a
```

```
rpm -q kernel
```

Where are you on the network? What connections are established?

```
ifconfig -a
```

```
netstat -antup
```

```
iptables -L
```

```
arp -e
```

What is running?

There are numerous local privilege escalation exploits out there in the void. Are there any vulnerable applications or services running that have known exploits?

Which services are being run with root privileges?

```
ps -ef | grep root
```

or

```
ps aux | grep root
```

```
cat /etc/services
```

Any vulnerable applications?

```
ls -alh /usr/bin/
```

```
ls -alh /sbin/
```

Any files with SUID/SGID permissions?

```
find / -perm -g=s -o -perm -u=s -type f 2>/dev/null
```

or, for a faster search in “bin” directories

```
for i in `locate -r "bin$"`; do find $i \( -perm -4000 -o -perm -2000 \) -ty
```

Uploading and running exploit code

If there is a local privilege escalation exploit available, how will you upload and execute the exploit code on your target?

What languages are supported on the machine?

```
find / -name 'language'
```

ex: find / -name python*

Is GCC present?

```
find / -name gcc
```

How can you upload the exploit code? Use `find` to look for things like:

wget, nc, netcat, tftp, ftp, fetch etc.

Where can you write and execute files?

You will need to find a place to compile and execute your exploit code

This will locate world writeable and world executable folders

```
find /\(-perm -o w -perm -o x\) -type d 2>/dev/null
```

Cracking password hashes

Can you view /etc/passwd and /etc/shadow ?

```
cat /etc/passwd
```

```
cat /etc/shadow
```

If you can, try to crack the hashes you find. You never know!

Limited Shell?

Give these a shot.

```
python -c 'import pty;pty.spawn("/bin/bash")
```

```
echo os.system('/bin/bash')
```

The simplest things are often overlooked

If I am ever stuck getting root privileges, its 9 times out of 10 because I am overthinking it. Sometimes the answer is so simple that its easy to overlook it. If you’re getting stuck, think back to square one and move forward slowly and pay attention to the details. Here are some of simple things that can be overlooked:

Is the account you are using a sudoer? If you have the password for the account, you may be able to use sudo. I have seen many people look over this. Are there other users that are sudoers?

```
cat /etc/sudoers
```

```
sudo -l
```

Always check for password reuse. Unless of course you don't want to be noisy and risk a failed authentication.

Sometimes people don't think straight and put plain text passwords in .txt files and spreadsheets and all kinds of terribly insecure places, so don't disregard the idea.

Enumeration scripts

To make life easier, you can write your own bash script to run whatever commands you want, although sometimes it is not plausible to do so. For example if you do not have the ability to upload files or execute shell scripts.

Anyways, those are my usual go-to commands when I start enumerating for priv-esc. It is not an exhaustive list by any means, there is a whole world of possibilities out there for getting root and hopefully this will help! I'm sure I will add things as I think of them, seeing as I wrote this on my lunch at work and probably forgot a bunch of stuff. Happy Hunting.

Thank you to g0tmi1k for your awesome post on the subject. I have used it more than any other blog I can think of!

19 ♡

The Ultimate Privilege Escalation Reference - [Wiki] 4

HackTheBox Irked writeup 2

created	last reply	replies	views	users	likes	links
May '17	Jan 2	22	13.2k	17	69	11

Sirius First 100 May '17

g0tmi1k's blog is dope, for window's privesc I usually use this as reference.
http://www.fuzzysecurity.com/tutorials/16.html 277

2 Replies ⌄ 2 ♡

oaktree Programmer and Part-Time Savage May '17

Hi,

Nice checklist you got here… on this site, though, we format code this way:

```
code
```

Not like

> **this**

Put ```*language* on top of the code and then ``` on the line under.

Thank you.

1 ♡

g0jirasan gojirasan May '17

Gotcha, just realized that. Ill fix it

♡

g0jirasan gojirasan ↪ Sirius May '17

That one is pretty amazing too. I used it for the OSCP labs a couple times.

operatorequals John May '17

Hey guys,
I have created this tool: https://github.com/operatorequals/gatheros 173
just to cope with Info Gathering in team environments (CTFs, etc).
There is also a blog post that describes its usage.

1 Reply 6

9 MONTHS LATER

CLOSED JAN 21, '18

This topic was automatically closed after 30 days. New replies are no longer allowed.

10 MONTHS LATER

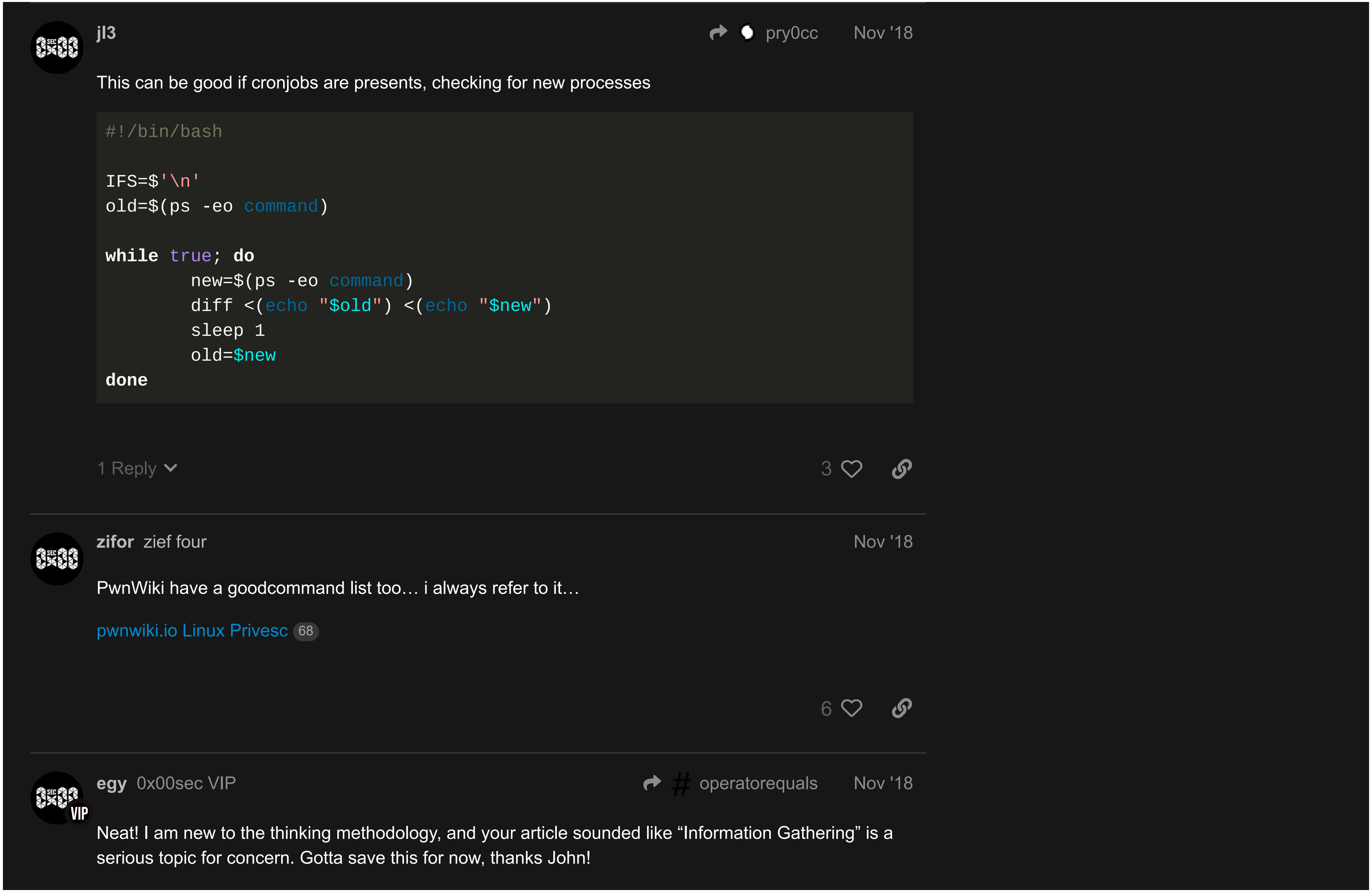
pry0cc Leader & Offsec Engineer & Forum Daddy Nov '18

Necrobumping this because it’s a really good article.

Anybody got any more thoughts on this? Do we have any new methods?

1 Reply 1

OPENED NOV 23, '18



jl3 pry0cc Nov '18

This can be good if cronjobs are presents, checking for new processes

```
#!/bin/bash

IFS=$'\n'
old=$(ps -eo command)

while true; do
        new=$(ps -eo command)
        diff <(echo "$old") <(echo "$new")
        sleep 1
        old=$new
done
```

1 Reply 3

zifor zief four Nov '18

PwnWiki have a goodcommand list too… i always refer to it…

pwnwiki.io Linux Privesc 68

6

egy 0x00sec VIP operatorequals Nov '18

Neat! I am new to the thinking methodology, and your article sounded like “Information Gathering” is a serious topic for concern. Gotta save this for now, thanks John!

1 ♡

mdeez Matt Nov '18

A while back I stumbled across a great enumeration script that hits most of the points brought up here. Take a look at https://github.com/rebootuser/LinEnum 29. This is the first thing I try to run on a fresh low-priv shell.

2 ♡

Nitrax 1337 0f The m0n7h Nov '18

Linuxprivchecker 25 could also be a good alternative to traditionnal tools. I revamped the original one to make it works with all python versions.

Hope it helps,
Best,
Nitrax

7 ♡

jl3 Nov '18

That’s fucking awesome. Thanks man

♡

iodbh IDdbH Nov '18

Thanks, great info !

Jake William's Wild West Hacking Fest 2018 talk "Privilege Escalation FTW" was relased a week ago, it has some good stuff too (Linux and Windows) :

4 ♡

pry0cc Leader & Offsec Engineer & Forum Daddy Nov '18

Just found this, too:

If you're looking for WIndows privesc, this is nice.

absolomb.com 40

Windows Privilege Escalation Guide

Privilege escalation always comes down to proper enumeration. But to accomplish proper enumeration you need to know what to check and look for. This takes familiarity with systems that normally comes along with experience. At first privilege...

5

AlexiBesto Sirius Nov '18

Yes, this is also very detailed. Thank you

The Ultimate Privilege Escalation Reference - [Wiki] 1

guly 0x00sec VIP jl3 Dec '18

pspy does this in a more efficient way for not really-hardened boxes

GitHub 32

DominicBreuker/pspy

Monitor linux processes without root permissions. Contribute to DominicBreuker/pspy development by creating an account on GitHub.

for CTF/boot2root, i also check file time using “user.txt” as a reference

3

Techno_Forg Zain Dec '18

Yo! Nice tut (?), but just curious if there’s a possibility for another one except for windows? Just a thought.

Also, I might put some of these commands in a bash script for automation of prosperity. 😛

Again, awesome tut! This will be helpful for HTB. ~Cheers!

–Techno Forg–