Lab – Exploit Vulnerable Web Applications Using Command Injection

Overview

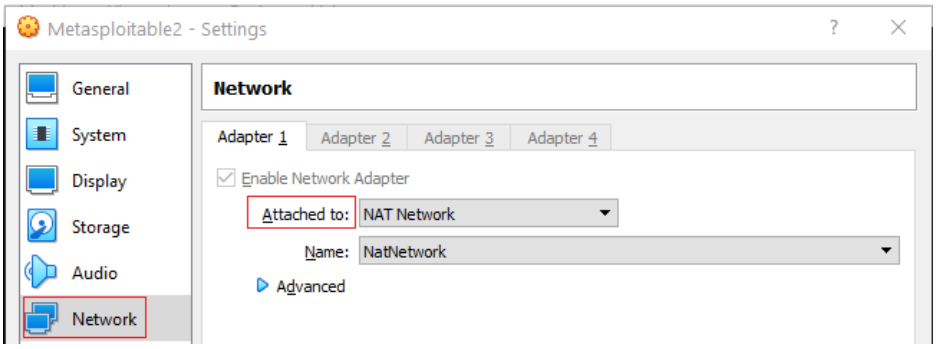
In this lab, you will learn how to exploit a vulnerable web application using command injection. Command injection is also known as OS Command injection, is an attack technique used to execute commands on a host operating system via a vulnerable web application.

Command Injection attacks are possible when an application passes unsafe user-supplied data (forms, cookies, HTTP headers, and so on) to a system shell. These commands are executed with the privileges of the vulnerable application. These attacks are due to the web application not having sufficient input validation on the command being run.

Lab Requirements

- Install of VirtualBox
- One virtual install of Kali Linux
- One virtual install of Metasploitable2

Ensure your VirtualBox network settings are set to NAT Network.



For this to work, we will need to have both Kali and Metasploitable2 up and running.

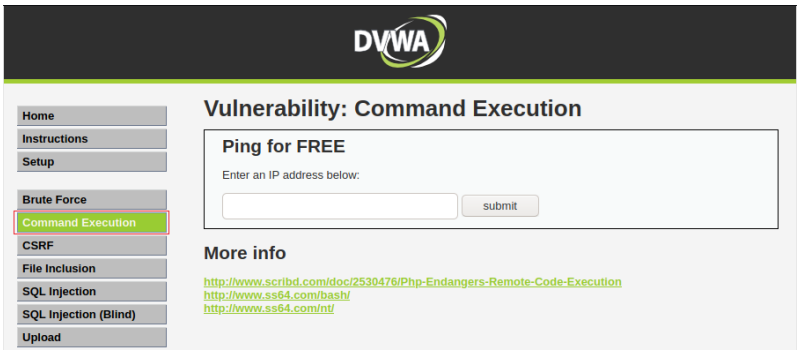
You will first need to log on to Metasploitable2 using the username and password of msfadmin. Once you log on, find the IP address assigned to Metasploitable2 using the `ifconfig` command. This is my IP address; yours will differ.

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f6:69:30
          inet addr:10.0.2.11  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fef6:6930/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:97 errors:0 dropped:0 overruns:0 frame:0
          TX packets:136 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18420 (17.9 KB)  TX bytes:63396 (61.9 KB)
          Base address:0xd020 Memory:f0200000-f0220000
```

Secondly, the security settings for Metasploitable2 must be set too low to ensure this lab will work. You first need to open your Kali web browser. In the address bar, type the IP address of your virtual install of Metasploitable2. This will open the DVWA home page.

From the menu on the left, select the DVWA Security option. From the main window, reduce the security level from high to low. Click the submit button.

With the DVWA home page open and the security set to low, click on the Command Execution link from the menu on the left.



Metacharacters

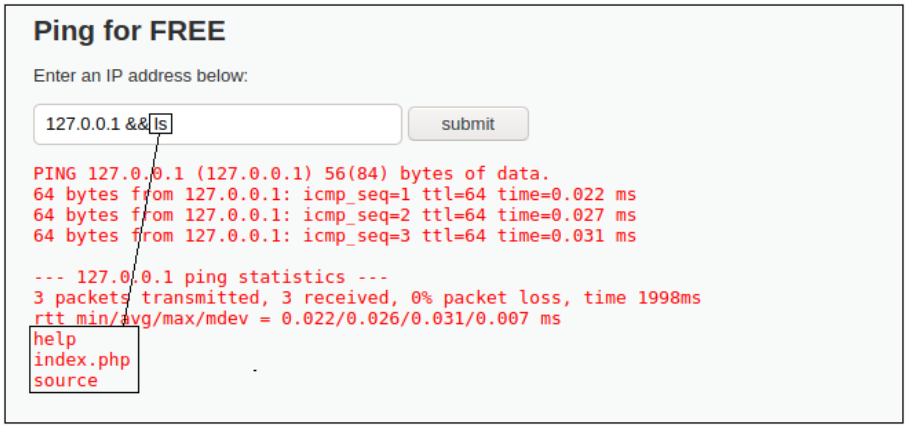
- **;** The semicolon is the most common metacharacter used to test an injection flaw. The shell will run all the commands in sequence separated by the semicolon.
- **&** Separate multiple commands on one command line. It runs the first command then the second command.
- **&&** Runs the command following && only if the preceding command is successful.
- | The Pipe pipes the output of the first command into the second command.
- || Redirects the standard outputs of the first command to standard input of the second command.

- ‘ The quote is used to force the shell to interpret and run commands between backticks. Following is an example of this command: Variable=”OS version ‘uname -a'” && echo $variable.
- **( )** The brackets are used to nest commands.
- # The Hash is used as a command-line comment.

Let us begin by pinging our target machine from the DWWA command execution page. If the ping command is successful, by appending the ls as an additional command using && to our initial command, the additional command should complete successfully as well.

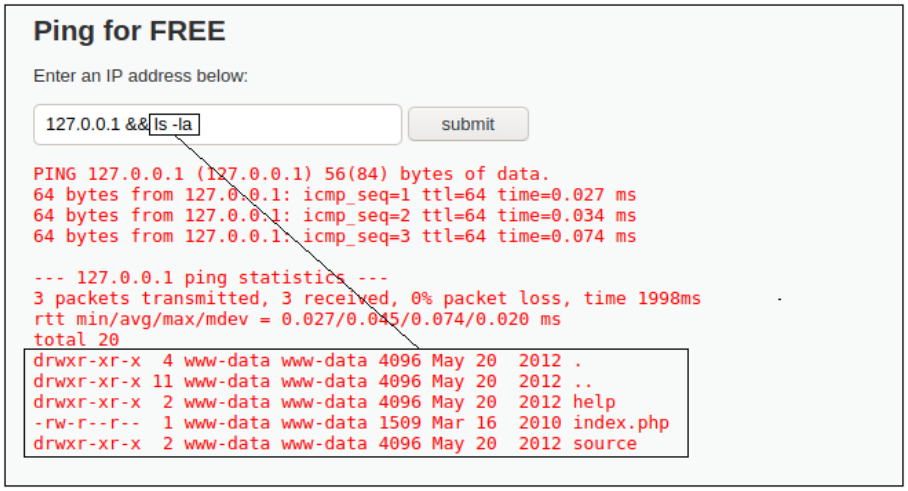
```
127.0.0.1 && ls
```

The **ls** command lists the names of the files and folders within the directory.



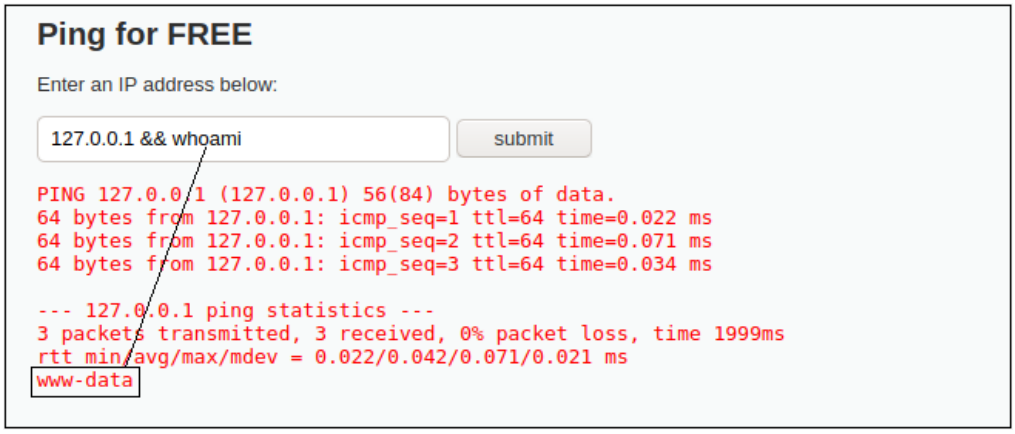
As the web application interacts with the operating system’s backend and is not sanitizing our input, we can introduce Metacharacters to string extra commands, allowing us to break out of its intended ping command and run our own commands directly on the backend operating system.

```
127.0.0.1 && ls -la
```



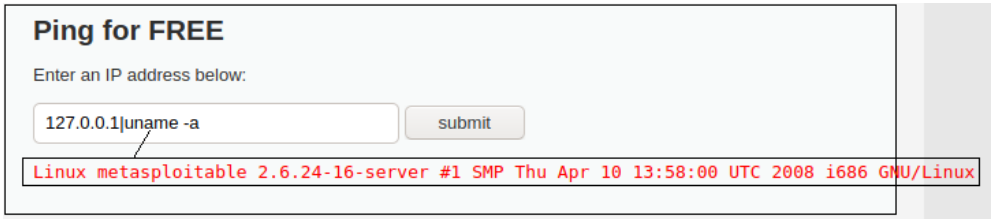
Here we are shown the file permissions of the directory for the www-data account. ‘www-data’ is the user under which the webserver runs. ‘www-data’ user has no password set by default.

```
127.0.0.1 && whoami
```



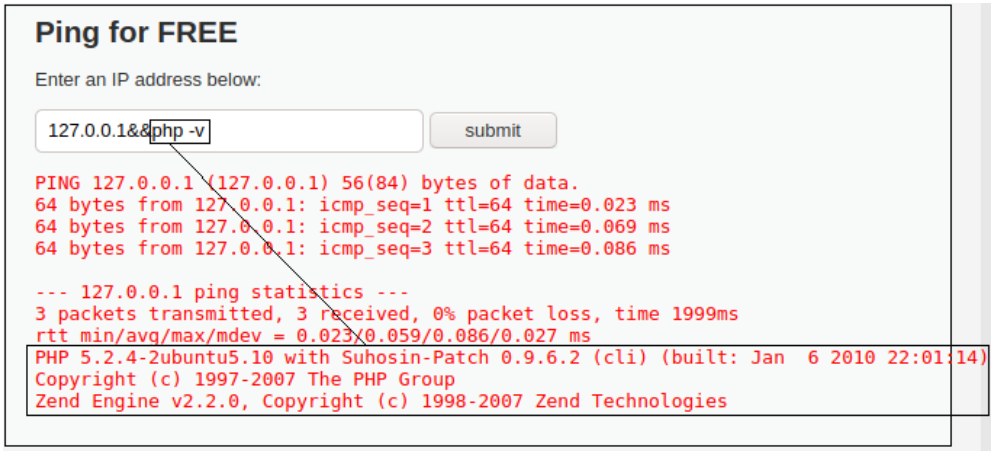
127.0.0.1 && whoami Shows you the user the web application is currently running as.

```
127.0.0.1|uname -a
```



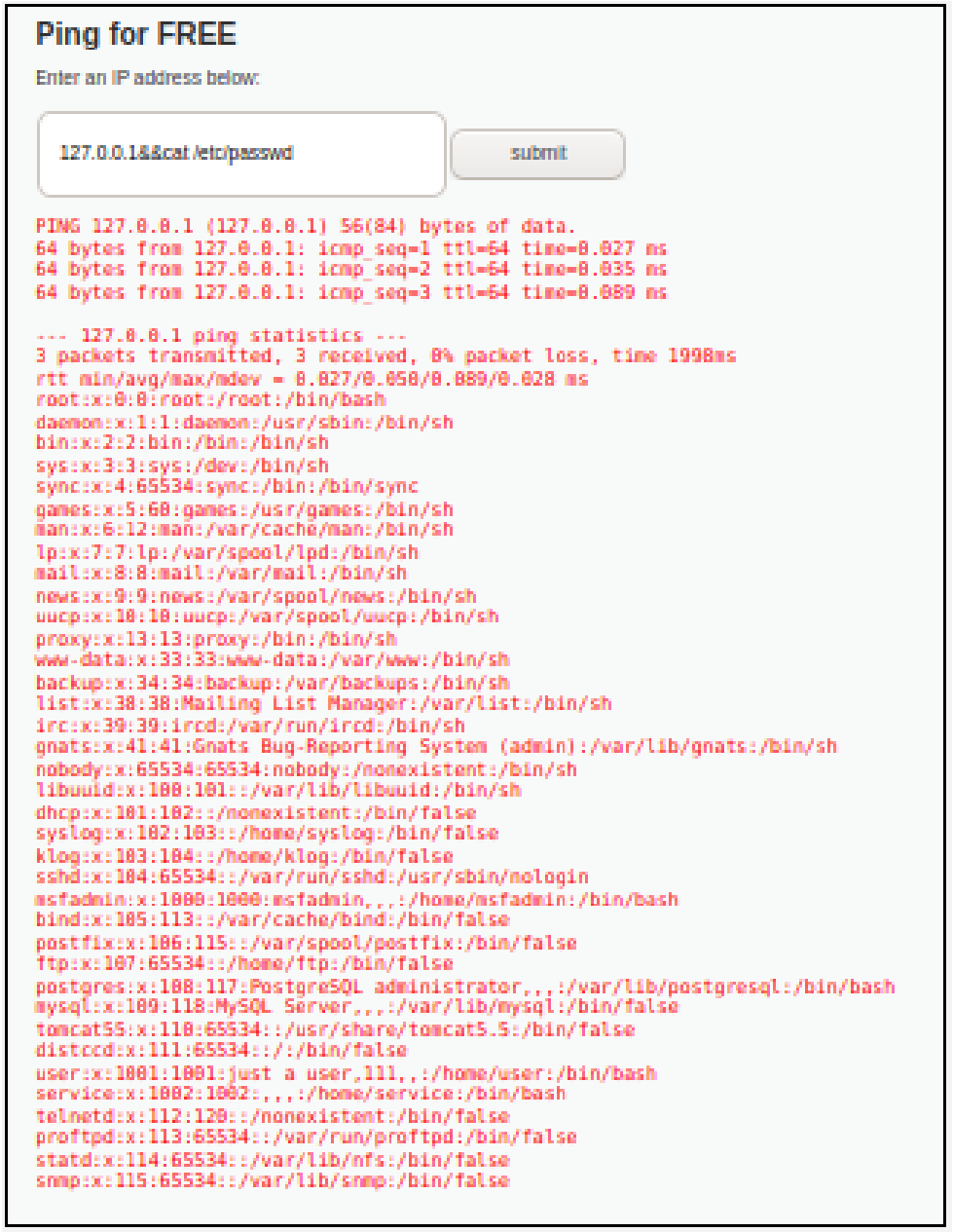
127.0.0.1|uname -a shows the Operating System version the webserver is running.

```
127.0.0.1&&php -v
```



127.0.0.1&&php -v Gives you PHP version running on web applications server.

127.0.0.1&&cat /etc/passwd



127.0.0.1&&cat /etc/passwd displays all the users on the backend Linux Server

Summary –

In this short lab, we learned how to exploit a web application using command injection.