



S3/L2

E S E R C I Z I O W E B A P P L I C A T I O N

TRACCIA



Nella lezione pratica di oggi vedremo come configurare una DVWA – ovvero damn vulnerable web application in Kali Linux. La DVWA ci sarà molto utile per i nostri test. DVWA è una web application scritta in PHP e MySQL installabile in qualsiasi ambiente in cui sia presente un web server, php e mysql.

L'applicazione è stata creata e concepita piena di vulnerabilità più o meno facili da scovare, il livello di difficoltà può essere configurato come:

1. Basso – Non esiste nessun tipo di controllo di sicurezza
2. Medio – Controlli approssimativi
3. Alto – Questo è il livello più alto e l'obiettivo non si deve sempre focalizzare soltanto sulla vulnerabilità stessa (stile CTF)
4. Impossibile – Non sono presenti vulnerabilità. Questo livello è stato creato per mostrare agli sviluppatori come mitigare le vulnerabilità.

Passaggio 1

Completare l'installazione di DVWA.

Dopo aver completato l'installazione, configuriamo all'interno del file config.inc.php utente e password.

```
(kali@kali)-[~]  
$ cd /var/www/html
```

```
(kali@kali)-[/var/www/html/DVWA/config]  
$ sudo cp config.inc.php.dist config.inc.php  
  
(kali@kali)-[/var/www/html/DVWA/config]  
$ sudo nano config.inc.php
```

```
1 <?php  
2  
3 # If you are having problems connecting to the MySQL database and all of the variables below are correct  
4 # try changing the 'db_server' variable from localhost to 127.0.0.1. Fixes a problem due to sockets.  
5 # Thanks to @digininja for the fix.  
6  
7 # Database management system to use  
8 $DBMS = 'MySQL';  
9 $DBMS = 'PGSQL'; // Currently disabled  
10  
11 # Database variables  
12 # WARNING: The database specified under db_database WILL BE ENTIRELY DELETED during setup.  
13 # Please use a database dedicated to DVWA.  
14 #  
15 # If you are using MariaDB then you cannot use root, you must use create a dedicated DVWA user.  
16 # See README.md for more information on this.  
17 $_DVWA = array();  
18 $_DVWA['db_server'] = getenv('DB_SERVER') ?: '127.0.0.1';  
19 $_DVWA['db_database'] = 'dvwa';  
20 $_DVWA['db_user'] = 'admin';  
21 $_DVWA['db_password'] = 'password';  
22 $_DVWA['db_port'] = '3306';  
23  
24 # ReCAPTCHA settings  
25 # Used for the 'Insecure CAPTCHA' module  
26 # You'll need to generate your own keys at: https://www.google.com/recaptcha/admin  
27 $_DVWA['recaptcha_public_key'] = '';  
28 $_DVWA['recaptcha_private_key'] = '';  
29  
30 # Default security level  
31 # Default value for the security level with each session.  
32 # The default is 'impossible'. You may wish to set this to either 'low', 'medium', 'high' or impossible'.  
33 $_DVWA['default_security_level'] = 'impossible';  
34  
35 # Default locale  
36 # Default locale for the help page shown with each session.  
37 # The default is 'en'. You may wish to set this to either 'en' or 'zh'.  
38 $_DVWA['default_locale'] = 'en';  
39  
40 # Disable authentication  
41 # Some tools don't like working with authentication and passing cookies around  
42 # so this setting lets you turn off authentication.  
43 $_DVWA['disable_authentication'] = false;  
44  
45 define('MYSQL', 'mysql');
```


```
(kali@kali)-[/var/www/html]  
$ sudo git clone https://github.com/digininja/DVWA  
Cloning into 'DVWA' ...  
remote: Enumerating objects: 4503, done.  
remote: Counting objects: 100% (53/53), done.  
remote: Compressing objects: 100% (44/44), done.  
remote: Total 4503 (delta 19), reused 33 (delta 8), pack-reused 4450  
Receiving objects: 100% (4503/4503), 2.38 MiB | 5.80 MiB/s, done.  
Resolving deltas: 100% (2114/2114), done.
```


Passaggio 2

Andiamo a settare il servizio apache2, modificando il file php.ini e consentendo le voci "allow_url_fopen" e "allow_url_include"

```
(kali@kali)-[~]  
$ cd /etc/php/8.2/apache2
```

```
(kali@kali)-[/etc/php/8.2/apache2]  
$ sudo mousepad php.ini  
[sudo] password for kali:
```



```
864 ; Whether to allow the treatment of URLs (like http:// or ftp://) as files.  
865 ; https://php.net/allow-url-fopen  
866 allow_url_fopen = On  
867 |  
868 ; Whether to allow include/require to open URLs (like https:// or ftp://) as files.  
869 ; https://php.net/allow-url-include  
870 allow_url_include = On  
871
```

Passaggio 3

Accediamo alla DVWA tramite 127.0.0.1/DVWA e
cambiamo il livello di sicurezza in “low”

A questo punto possiamo aprire Burpsuite, e
provare ad intercettare la richiesta di login.


DVWA Security

Security Level

Security level is currently: **low**.

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.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.
Prior to DVWA v1.9, this level was known as 'high'.

Low 

Submit

Security level set to low



```
Pretty  Raw  Hex
1 POST /DVWA/login.php HTTP/1.1
2 Host: 127.0.0.1
3 Content-Length: 88
4 Cache-Control: max-age=0
5 sec-ch-ua: "Chromium";v="121", "Not A{Brand";v="99"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Linux"
8 Upgrade-Insecure-Requests: 1
9 Origin: http://127.0.0.1
10 Content-Type: application/x-www-form-urlencoded
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
12 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Referer: http://127.0.0.1/DVWA/login.php
18 Accept-Encoding: gzip, deflate, br
19 Accept-Language: en-US,en;q=0.9
20 Cookie: PHPSESSID=32vl6jrhv3dhmf2o5hhi88nmfd; security=low
21 Connection: close
22
23 username=admin&password=password&Login=Login&user_token=f265a9d2dc64493fa3a878167843806a
```

Passaggio 4

Dopo aver modificato password e username in "ciao" e "ciao", proviamo ad effettuare nuovamente un login.

Il login non potrà avvenire e ci verrà rilasciato un messaggio di errore.



Username

Password

Login

Login failed

```
Response
Pretty Raw Hex Render
49 <label for="pass">
    Password
</label>
<input type="password" class="loginInput" AUTOCOMPLETE="off" size="20" name="password">
<br />
50
51 <br />
52
53 <p class="submit">
    <input type="submit" value="Login" name="Login">
</p>
54
55 </fieldset>
56
57 <input type='hidden' name='user_token' value='a7e063d4be6dda25055891b8792b768e' />
58
59 </form>
60
61 <br />
62
63 <div class="message">
    Login failed
</div>
64
65 <br />
66 <br />
67 <br />
68 <br />
69 <br />
70 <br />
71 <br />
72 <br />
73
74 </div>
75 <!--<div id="content">-->
76
77 <div id="footer">
78
79 <p>
80 <a href="https://github.com/digininja/DVWA/" target="_blank">
    Damn Vulnerable Web Application (DVWA)
</a>
</p>
</div>
```