

S3-L2

Per la lezione pratica di oggi, l'obiettivo è quello di configurare una DVWA (Damn Vulnerable Web Application) in Kali Linux.

Ci assicuriamo di effettuare correttamente il setup delle risorse che andremo ad utilizzare.

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

```
(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), 2.30 MiB | 5.80 MiB/s, done.  
Resolving deltas: 100% (2114/2114), done.
```

```
(kali㉿kali)-[/var/www/html]  
$ sudo chmod -R 777 DVWA/  
  
(kali㉿kali)-[/var/www/html]  
$ cd DVWA/config
```

```
(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
```

?php

If you are having problems connecting to the MySQL database **and** all of the variables below are correct **try** changing the '**db_server**' variable from localhost to 127.0.0.1. Fixes a problem due to sockets.

Thanks to @digininja **for** the fix.

Database management system to **use**

DBMS = 'MySQL';

\$DBMS = 'PGSQL'; // **Currently disabled**

Database variables

WARNING: The database specified under db_database WILL BE ENTIRELY DELETED during setup.

Please **use** a database dedicated to DVWA.

If you are using MariaDB then you cannot **use** root, you must **use** create a dedicated DVWA user.

See README.md **for** more information on this.

_DVWA = array();

_DVWA['**db_server**'] = getenv('**DB_SERVER**') ?: '127.0.0.1';

_DVWA['**db_database**'] = 'dvwa';

_DVWA['**db_user**'] = 'kali';

_DVWA['**db_password**'] = 'kali';

_DVWA['**db_port**'] = '3306';

ReCAPTCHA settings

Used **for** the '**Insecure CAPTCHA**' module

You'll need to generate your own keys at: <https://www.google.com/recaptcha/admin>

_DVWA['**recaptcha_public_key**'] = '';

_DVWA['**recaptcha_private_key**'] = '';

Default security level

Default value **for** the security level with each session.

The **default** is '**impossible**'. You may wish to set this to either '**low**', '**medium**', '**high**' or impossible'.

```
(kali㉿kali)-[~]  
$ sudo service mysql start  
[sudo] password for kali:  
  
(kali㉿kali)-[~]  
$ sudo mysql -u root -p  
Enter password:  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.11.6-MariaDB-2 Debian n/a  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement
```

```
(kali㉿kali)-[~]  
$ sudo mysql -u root -p  
Enter password:  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.11.6-MariaDB-2 Debian n/a  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement  
.  
  
MariaDB [(none)]> create user 'kali'@'127.0.0.1' identified by 'kali';  
Query OK, 0 rows affected (0.009 sec)  
  
MariaDB [(none)]> grant all privileges on dvwa.* to 'kali'@'127.0.0.1' identi  
fied by 'kali';  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual th  
at corresponds to your MariaDB server version for the right syntax to use nea  
r 'privileges on dvwa.* to 'kali'@'127.0.0.1' identified by 'kali' at line 1  
MariaDB [(none)]> grant all privileges on dvwa.* to 'kali'@'127.0.0.1' identified by 'ka  
li';  
Query OK, 0 rows affected (0.002 sec)
```

```
(kali㉿kali)-[~]
$ service apache2 start

(kali㉿kali)-[~]
$ cd /etc/php

(kali㉿kali)-[/etc/php]
$ ls -l
total 4
drwxr-xr-x 5 root root 4096 Feb 25 10:47 8.2

(kali㉿kali)-[/etc/php]
$ ls
8.2

(kali㉿kali)-[/etc/php]
$ cd /etc/php/8.2/apache2

(kali㉿kali)-[/etc/php/8.2/apache2]
$
```

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

```
; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; https://php.net/allow-url-fopen
allow_url_fopen = On

; Whether to allow include/require to open URLs (like https:// or ftp://) as files.
; https://php.net/allow-url-include
allow_url_include = On
```

Dopo esserci assicurati di avere tutto il necessario andiamo ad impostare il livello di sicurezza della nostra DVWA come bassa.

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

Cambiamo le impostazioni di gestione delle redirection responses del repeater, permettendogli di processare i cookies in redirects.

Seguendo le istruzioni intercettiamo un tentativo di login nella nostra DVWA, andiamo poi a modificare le credenziali in modo che non siano corrette.

⚡ Burp Suite Community Edition v2024.2.1.5 - Temporary Project

Burp Project Intruder Repeater View Help

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history Proxy settings

✎ Request to http://127.0.0.1:80

Forward Drop **Intercept is on** Action Open browser

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="123", "Not:A-Brand";v="8"
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/123.0.6312.122 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: security=low; PHPSESSID=dhe0q8ur1rmjgdiorfchhtmkrq
21 Connection: close
22
23 username=nongiusto&password=sbagliato&Login=Login&user_token=66c4afe9b90dee978358b142cea0547a
```

Inviando la richiesta modificata al repeater, la inviamo alla DVWA e ne seguiamo la redirection. Com'è possibile vedere in figura il login non è andato a buon fine, il nostro esperimento ha quindi avuto successo.

Burp Suite Community Edition v2024.2.1.5 - Temporary Project

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

1 x 2 x 3 x 4 x 5 x 6 x 7 x 8 x +

Send Cancel < >

Request

Pretty Raw Hex

```
1 GET /DWA/login.php HTTP/1.1
2 Host: 127.0.0.1
3 Cache-Control: max-age=0
4 sec-ch-ua: "Chromium";v="123", "Not:A-Brand";v="8"
5 sec-ch-ua-mobile: ?0
6 sec-ch-ua-platform: "Linux"
7 Upgrade-Insecure-Requests: 1
8 Origin: http://127.0.0.1
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
  Gecko) Chrome/123.0.6312.122 Safari/537.36
10 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
11 Sec-Fetch-Site: same-origin
12 Sec-Fetch-Mode: navigate
13 Sec-Fetch-User: ?1
14 Sec-Fetch-Dest: document
15 Referer: http://127.0.0.1/DWA/login.php
16 Accept-Encoding: gzip, deflate, br
17 Accept-Language: en-US,en;q=0.9
18 Cookie: security=low; PHPSESSID=bb3hdjnvkhuiccdqk8ndt5dv8r
19 Connection: close
20
21
```

Response

Pretty Raw Hex Render

```
54
55
56
57 <input type='hidden' name='user_token' value='3c13ff1369a3ee8c83b44123a30e9223' />
58
59 </form>
60
61 <br />
62
63 <div class="message">
64   Login failed
65 </div>
66
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/DWA/" target="_blank">
81       Damn Vulnerable Web Application (DVWA)
82     </a>
83   </p>
84
85 </div>
86 <!--<div id="footer"> -->
87
88 </div>
89 <!--<div id="wrapper"> -->
90
91 </body>
```

Possiamo infine vedere la traduzione grafica del codice HTML che ci mostra ancora una volta che il login è fallito.



Username

Password

Login

Login failed

Done by:

André Vinícius, Federico Biggi, Joel Landry, Lorenzo Franchi, Otman Hmich, Romano Cascialli, Samuele Aversa