

# Esercizio Web Application – preparazione ambiente S3/L2

Manuel Di Gangi

6 febbraio 2024

## 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.

1. Aprite un terminale su Kali, utilizzate l'utenza di root, eseguendo il comando «sudo su» e poi eseguite i comandi seguenti:
  - `cd /var/www/html`
  - `git clone`  
<https://github.com/digininja/DVWA>
  - `chmod -R 777 DVWA/`
  - `cd DVWA/config`
  - `cp config.inc.php.dist config.inc.php`
  - `nano config.inc.php`

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
└─$ sudo su
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
(kali@kali)-[~]
└─$ cd /var/www/html

(kali@kali)-[/var/www/html]
└─$ git clone https://github.com/digininja/DVWA
Cloning into 'DVWA' ...
remote: Enumerating objects: 4494, done.
remote: Counting objects: 100% (44/44), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 4494 (delta 15), reused 33 (delta 8), pack-reused 4450
Receiving objects: 100% (4494/4494), 2.29 MiB | 796.00 KiB/s, done.
Resolving deltas: 100% (2110/2110), done.

(kali@kali)-[/var/www/html]
└─$ chmod -R 777 DVWA/

(kali@kali)-[/var/www/html]
└─$ cd DVWA/config

(kali@kali)-[/var/www/html/DVWA/config]
└─$ cp config.inc.php.dist config.inc.php

(kali@kali)-[/var/www/html/DVWA/config]
└─$ nano config.inc.php

(kali@kali)-[/var/www/html/DVWA/config]
└─$ service mysql start

(kali@kali)-[/var/www/html/DVWA/config]
└─$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.11.5-MariaDB-3 Debian n/a
```

2. Facciamo partire il servizio mysql, connettiamoci al DB e creiamo un'utenza;

```
MariaDB [(none)]> create user 'kali'@'127.0.0.1' identified by 'kali' ;
Query OK, 0 rows affected (0.062 sec)
```

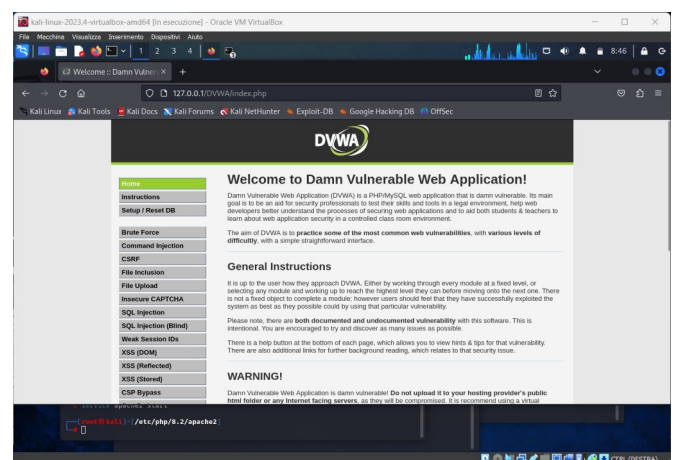
```
MariaDB [(none)]> grant all privileges on dvwa.* to 'kali'@'127.0.0.1' identified by 'kali';
Query OK, 0 rows affected (0.003 sec)
```

3. Facciamo partire il servizio apache (il web server) e modifichiamo le voci `allow_url_fopen` e `allow_url_include` come sotto

```
;;;;;;;;;;;;;;
; 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
```

4. Creiamo il database ed effettuiamo il login.



5. Ora che l'ambiente è creato, eseguiamo delle prove con Burpsuite.

Effettuiamo il login dal browser  
all'indirizzo 127.0.0.1/DVWA  
Utente: admin  
Password: password

Intercettiamo la richiesta del client al  
server e andiamo a modificare le  
credenziali di accesso

Utente: utente  
Password: spagliato

```
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="119", "Not?A_Brand";v="24"
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/119.0.6045.159 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=impossible; PHPSESSID=hi5ldoh0bppqoo5rj52afage5
21 Connection: close
22
23 username=admin&password=password&Login=Login&user_token=9de9cb862e795afaa6cf02beb3d40d23

username=utente&password=spagliato&Login=Login&user_token=9de9cb862e795afaa6cf02beb3d40d23
```



6. Come ci aspettavamo con le credenziali errate non riusciamo ad entrare.

Username  
utente

Password  
\*\*\*\*\*

Login

CSRF token is incorrect

Request

Pretty

Raw

Hex

1

GET /DVWA/login.php HTTP/1.1

2

Host: 127.0.0.1

3

Cache-Control: max-age=0

4

sec-ch-ua: "Chromium";v="119",

5

"Not?A\_Brand";v="24"

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

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.6045.159 Safari/537.36

11

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

12

Sec-Fetch-Site: same-origin

13

Sec-Fetch-Mode: navigate

14

Sec-Fetch-User: ?1

15

Referer: http://127.0.0.1/DVWA/login.php

16

Accept-Encoding: gzip, deflate, br

17

Accept-Language: en-US,en;q=0.9

18

Cookie: security=impossible; PHPSESSID=

19

h1SLdoh0Bppq05rj52afage5

20

Connection: close

21

Response

Pretty

Raw

Hex

Render

1

HTTP/1.1 200 OK

2

Date: Tue, 06 Feb 2024 14:29:11 GMT

3

Server: Apache/2.4.58 (Debian)

4

Expires: Tue, 23 Jun 2009 12:00:00 GMT

5

Cache-Control: no-cache, must-revalidate

6

Pragma: no-cache

7

Vary: Accept-Encoding

8

Content-Length: 1342

9

Connection: close

10

Content-Type: text/html; charset=utf-8

11

12

<!DOCTYPE html>

13

14

<html lang="en-GB">

15

16

<head>

17

18

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

19

20

<title>

21

Login :: Damn Vulnerable Web Application (DVWA)

22

</title>

23

24

<link rel="stylesheet" type="text/css" href="dvwa/css/login.css" />

25

</head>

26

<body>

27

28

<div id="wrapper">

29

30

<div id="header">

31

32

<br />

33

34