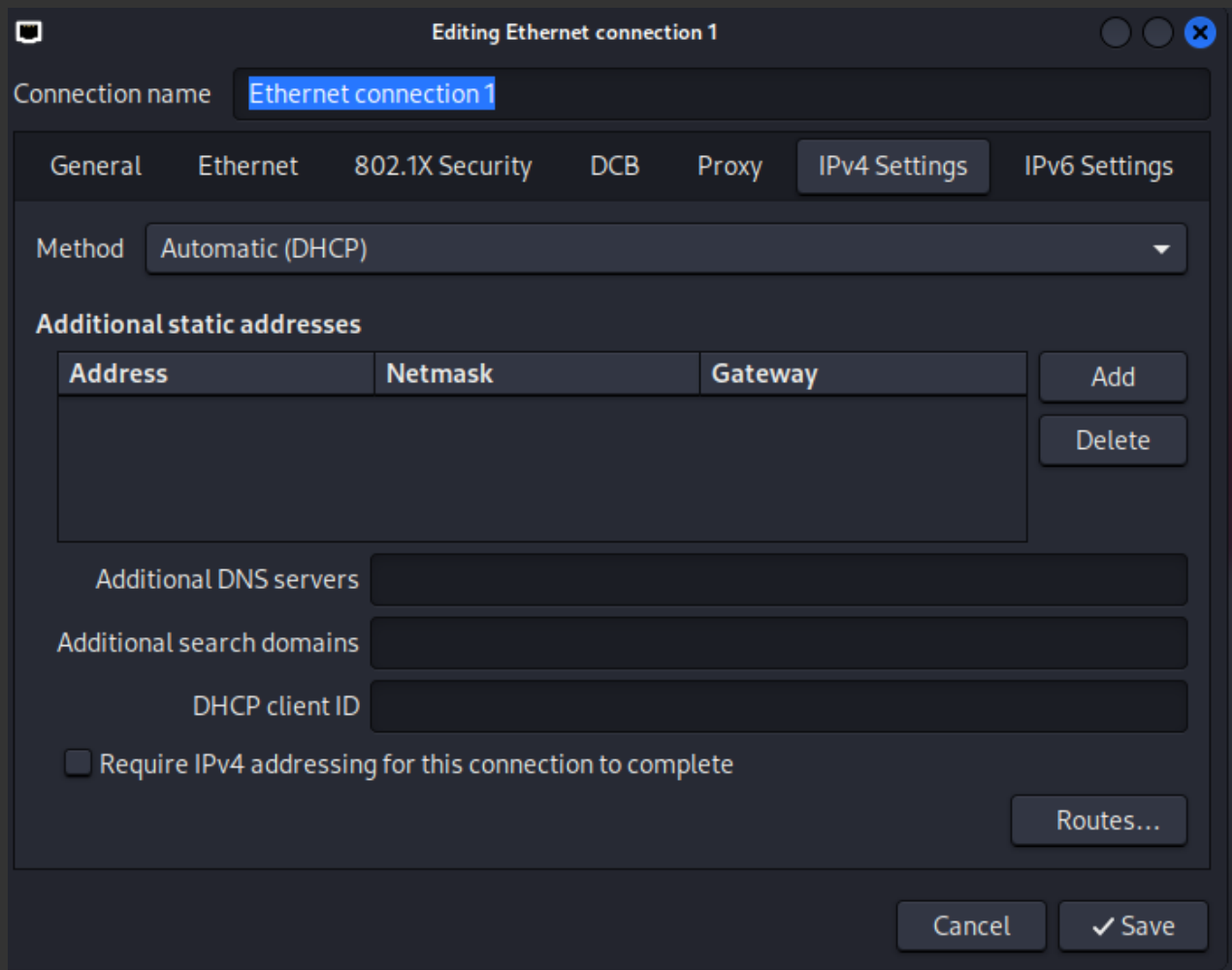
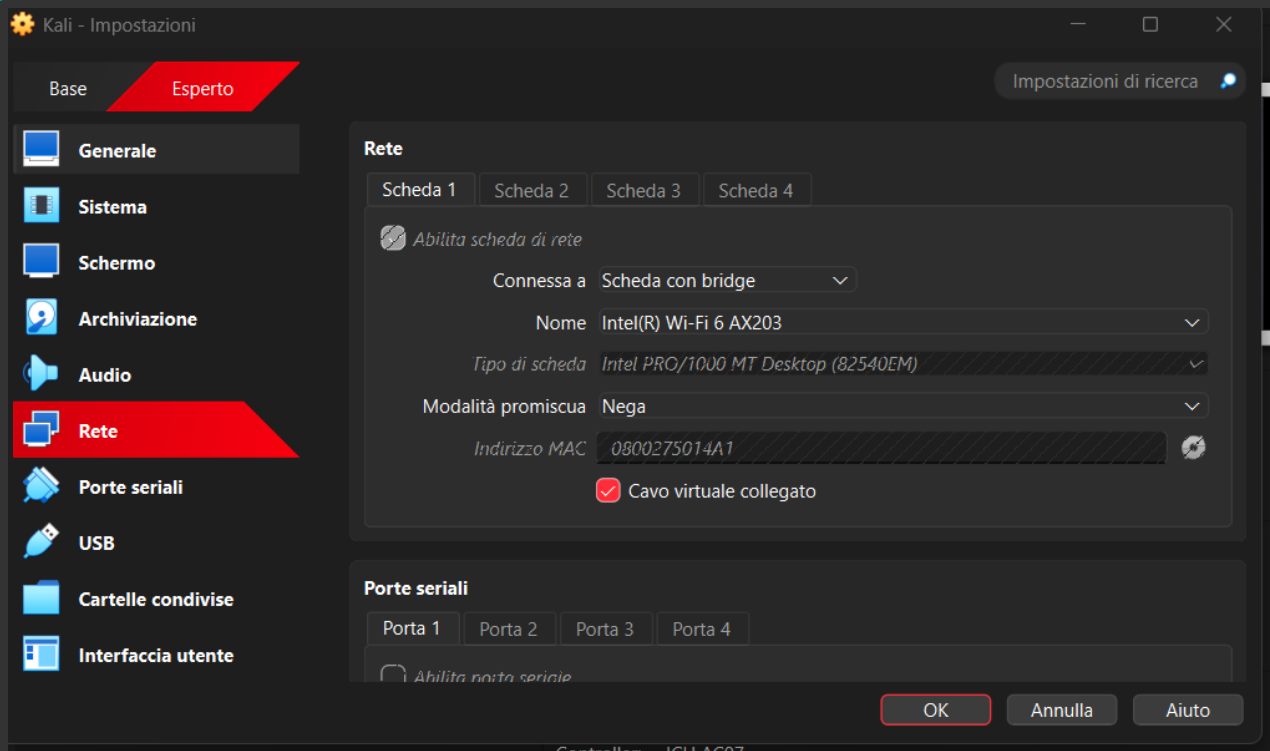


Installazione DVWA

Per prima cosa sono andato ad impostare la macchina con connessione a Scheda con bridge, in modo da poter comunicare con la rete esterna



Poi si andrà a scaricare ed installare la DVWA

```
(kali㉿kali)-[~]
$ sudo su
[sudo] password for kali:
(kali㉿kali)-[/home/kali]
# cd /var/www/html

(kali㉿kali)-[/var/www/html]
# git clone https://github.com/digininja/DVWA
Cloning into 'DVWA' ...
remote: Enumerating objects: 5611, done.
remote: Counting objects: 100% (76/76), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 5611 (delta 59), reused 37 (delta 37), pack-reused 5535 (from 3)
Receiving objects: 100% (5611/5611), 2.65 MiB | 24.00 KiB/s, done.
Resolving deltas: 100% (2784/2784), 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
```

```
<?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
$dbms = getenv('DBMS') ?: 'MySQL';
#$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'] = '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'] = getenv('RECAPTCHA_PUBLIC_KEY') ?: '';
$_DVWA['recaptcha_private_key'] = getenv('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

^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_/ Go To Line  M-E Redo
```

```

(kali㉿kali)-[~]
$ sudo su
[sudo] password for kali:
(kali㉿kali)-[/home/kali]
# service mysql start

(kali㉿kali)-[/home/kali]
# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 11.8.1-MariaDB-4 Debian n/a

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>

```

```

MariaDB [(none)]> create user 'kali'@'127.0.0.1' identified by 'kali'
→ ;
Query OK, 0 rows affected (0.054 sec)

MariaDB [(none)]> grant all privileges on dvwa.* to 'kali'@'127.0.0.1' identified by 'kali';
Query OK, 0 rows affected (0.044 sec)

MariaDB [(none)]> exit
Bye

(kali㉿kali)-[/home/kali]
#

```

```

(kali㉿kali)-[/home/kali]
# service apache2 start

(kali㉿kali)-[/home/kali]
# cd /etc/php/8.1/apache2
cd: no such file or directory: /etc/php/8.1/apache2

(kali㉿kali)-[/home/kali]
# cd /etc/php/

(kali㉿kali)-[/etc/php]
# ls
8.4

(kali㉿kali)-[/etc/php]
# cd 8.4

(kali㉿kali)-[/etc/php/8.4]
# cd apache2

(kali㉿kali)-[/etc/php/8.4/apache2]
# ls
conf.d  php.ini

(kali㉿kali)-[/etc/php/8.4/apache2]
# sudo nano php.ini

```

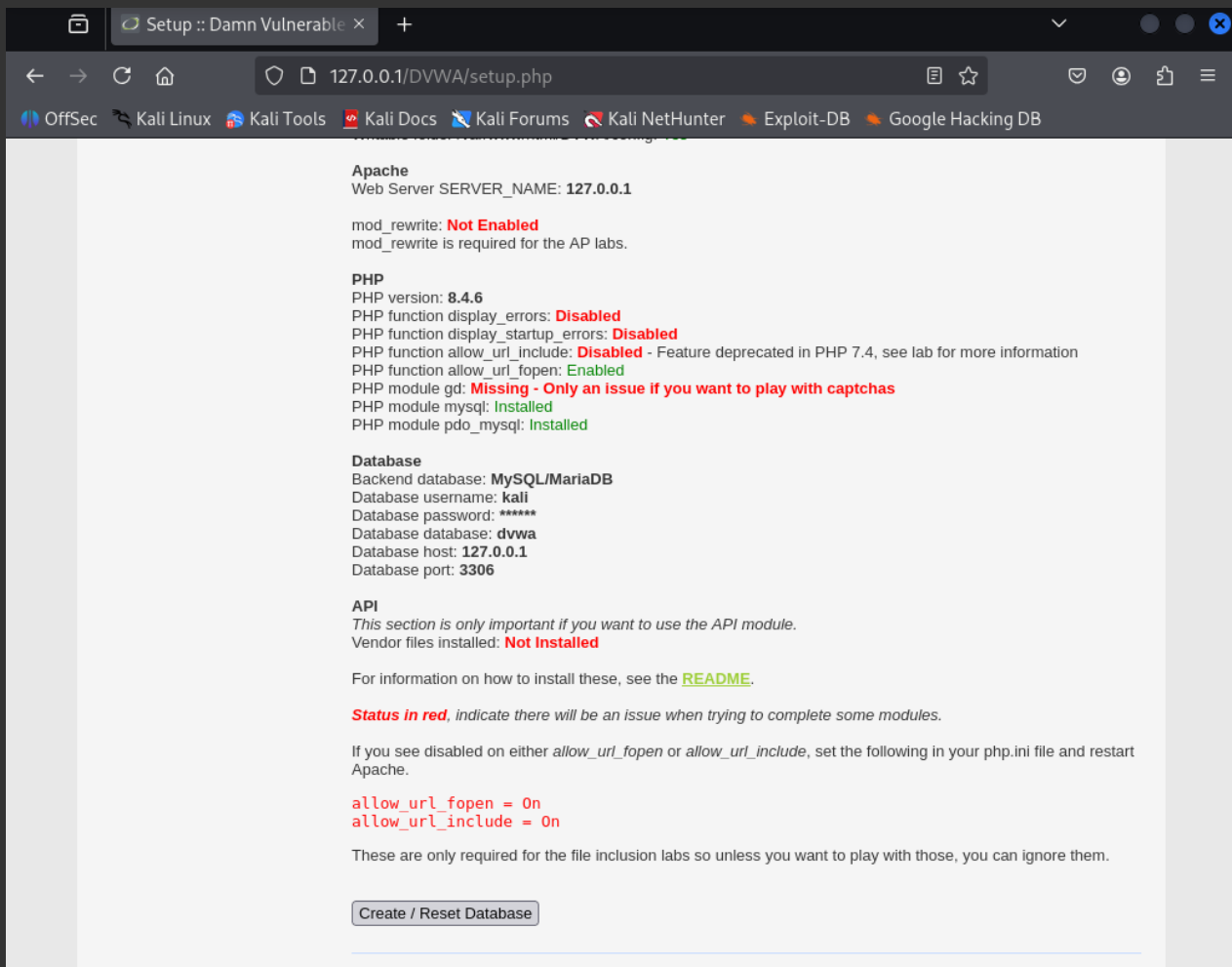
```

; 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

```

A questo punto cerchiamo nella barra degli indirizzi sul Browser 127.0.0.1/DVWA/setup.php e andiamo a creare u database, impostando poi la difficoltà su “low”



Security Level

Security level is currently: **impossible**.

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

A questo punto apriamo Burpsuite, apriamo un browser e facciamo l'accesso sulla DVWA. In questo caso si possono intercettare le richieste, e modificarne i pacchetti con la richiesta di accesso.

The screenshot shows the Burp Suite interface with the 'Intercept' tab selected. A request to `http://127.0.0.1/DVWA/login.php` is intercepted. The 'Request' tab is active, displaying the raw HTTP request. The 'Inspector' panel on the right shows the request details, including headers, query parameters, body parameters, cookies, and headers.

Request

```
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Linux"
8 Accept-Language: en-US,en;q=0.9
9 Origin: http://127.0.0.1
10 Content-Type: application/x-www-form-urlencoded
11 Upgrade-Insecure-Requests: 1
12 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36
13 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
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: http://127.0.0.1/DVWA/login.php
19 Accept-Encoding: gzip, deflate, br
20 Cookie: security=impossible; PHPSESSID=69ac511210dbbeb0339f1022df1dd982
21 Connection: keep-alive
22
23 username=admin&password=password&Login=Login&user_token=d4a29a438949e4c4baf5f232fa269e60
```

Inspector

- Request attributes: 2
- Request query parameters: 0
- Request body parameters: 4
- Request cookies: 2
- Request headers: 20

```
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Linux"
8 Accept-Language: en-US,en;q=0.9
9 Origin: http://127.0.0.1
10 Content-Type: application/x-www-form-urlencoded
11 Upgrade-Insecure-Requests: 1
12 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36
13 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
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: http://127.0.0.1/DVWA/login.php
19 Accept-Encoding: gzip, deflate, br
20 Cookie: security=impossible; PHPSESSID=69ac511210dbbeb0339f1022df1dd982
21 Connection: keep-alive
22
23 username=daniele&password=ptata&Login=Login&user_token=d4a29a438949e4c4baf5f232fa269e60
```

The screenshot shows the DVWA login page. The 'Username' and 'Password' fields are filled with 'daniele' and 'ptata' respectively. The 'Login' button is clicked, and the message 'Login failed' is displayed at the bottom.

DVWA

Username

Password

Login

Login failed