

# **Индивидуальный проект. Этап №2**

**Основы информационной безопасности**

Астраханцева А. А.

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение</b>	<b>6</b>
<b>4</b>	<b>Выводы</b>	<b>12</b>

## Список иллюстраций

3.1	Репозиторий github DVWA . . . . .	6
3.2	Клонирование репозитория . . . . .	7
3.3	Перемещение DVWA . . . . .	7
3.4	Запуск apache сервера . . . . .	8
3.5	Сообщение о конфигурационном файле . . . . .	8
3.6	Копирование конфигурационного файла . . . . .	9
3.7	Просмотр конфигурационного файла . . . . .	9
3.8	Запуск mariadb . . . . .	10
3.9	Создание новой базы данных . . . . .	10
3.10	Завершение установки . . . . .	11

# 1 Цель работы

Ознакомление с специально предназначенным для поиска уязвимостей веб приложением под названием Damn Vulnerable Web Application (DVWA).

## 2 Задание

Установка DVWA в гостевую систему к Kali Linux.

## 3 Выполнение

Переходим в репозиторий github DVWA и копируем ссылку, по которой в дальнейшем будет клонировать репозиторий (рис. 3.1).

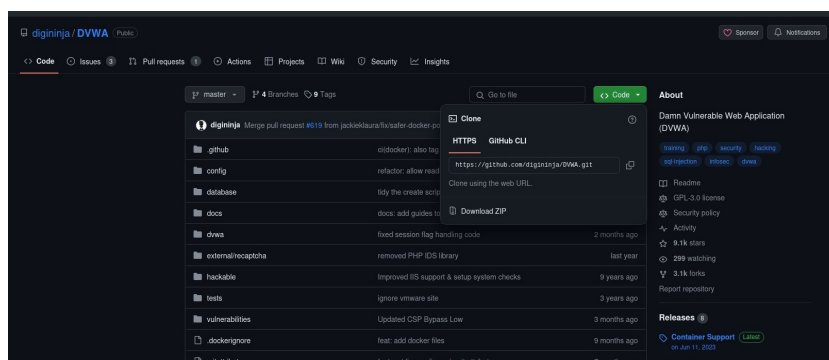
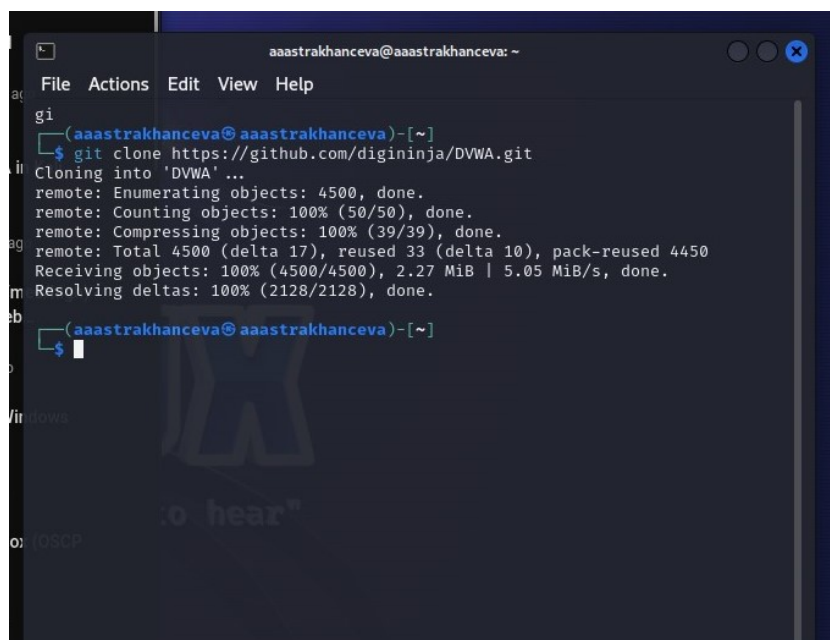


Рис. 3.1: Репозиторий github DVWA

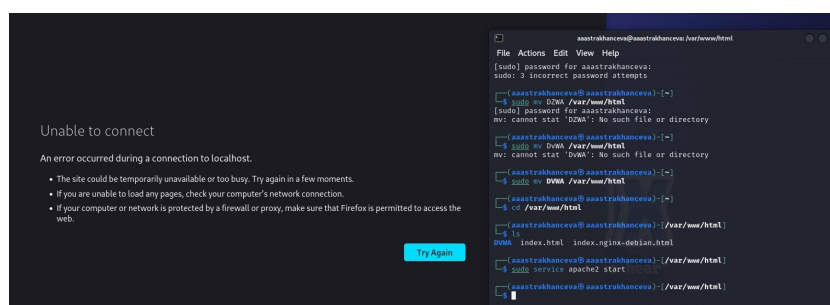
Клонируем репозиторий у себя в терминале (рис. 3.2).



```
aaastrakhanceva@aaastrakhanceva: ~  
File Actions Edit View Help  
gi  
$ git clone https://github.com/digininja/DVWA.git  
Cloning into 'DVWA' ...  
remote: Enumerating objects: 4500, done.  
remote: Counting objects: 100% (50/50), done.  
remote: Compressing objects: 100% (39/39), done.  
remote: Total 4500 (delta 17), reused 33 (delta 10), pack-reused 4450  
Receiving objects: 100% (4500/4500), 2.27 MiB | 5.05 MiB/s, done.  
Resolving deltas: 100% (2128/2128), done.  
$
```

Рис. 3.2: Клонирование репозитория

Перемещаем файл DVWA по в папку “/var/www/html” и проверяем, что файл успешно перемещен (рис. 3.3).



```
aaastrakhanceva@aaastrakhanceva: /var/www/html  
File Actions Edit View Help  
[sudo] password for aaastrakhanceva:  
sudo: 3 incorrect password attempts  
$ mv DVWA /var/www/html  
mv: cannot stat 'DVWA': No such file or directory  
$ mv DVWA /var/www/html  
mv: cannot stat 'DVWA': No such file or directory  
$ ls  
DVWA index.html index.nginx-debian.html  
$ sudo service apache2 start  
$
```

Рис. 3.3: Перемещение DVWA

После этого запускаем арасхе сервер и в браузерe открываем локальный сервер “http://localhost”. (рис. 3.4).

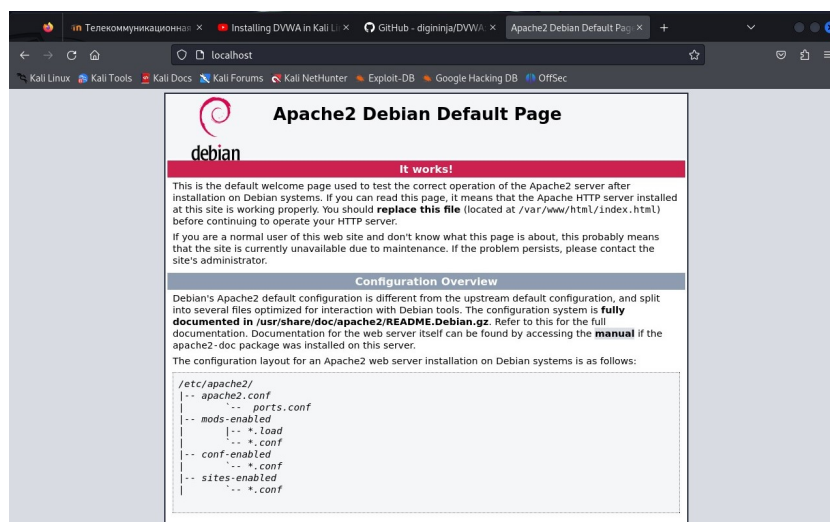


Рис. 3.4: Запуск apache сервера

Переходим по адресу “http://localhost/DVWA”, получаем сообщение о необходимости скопировать конфигурационный файл “config.inc.php.dist” в “config.inc.php”. (рис. 3.5).

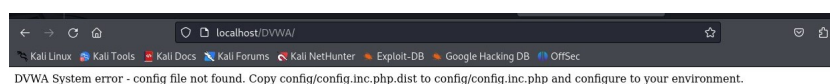


Рис. 3.5: Сообщение о конфигурационном файле

Вполняем копирование (рис. 3.6).



```
(aaastrakhanceva@aaastrakhanceva)-[/var/www/html]
$ cd DVWA

(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA]
$ ls
CHANGELOG.md  README.id.md  compose.yml  hackable  robots.txt
COPYING.txt   README.md     config       index.php  security.php
Dockerfile    README.pt.md  database     instructions.php  security.txt
README.ar.md  README.tr.md  docs        login.php  setup.php
README.es.md  README.zh.md  dvwa       logout.php  tests
README.fa.md  SECURITY.md   external    php.ini    vulnerabilities
README.fr.md  about.php    favicon.ico  phpinfo.php

(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA]
$ cd config

(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA/config]
$ ls
config.inc.php.dist

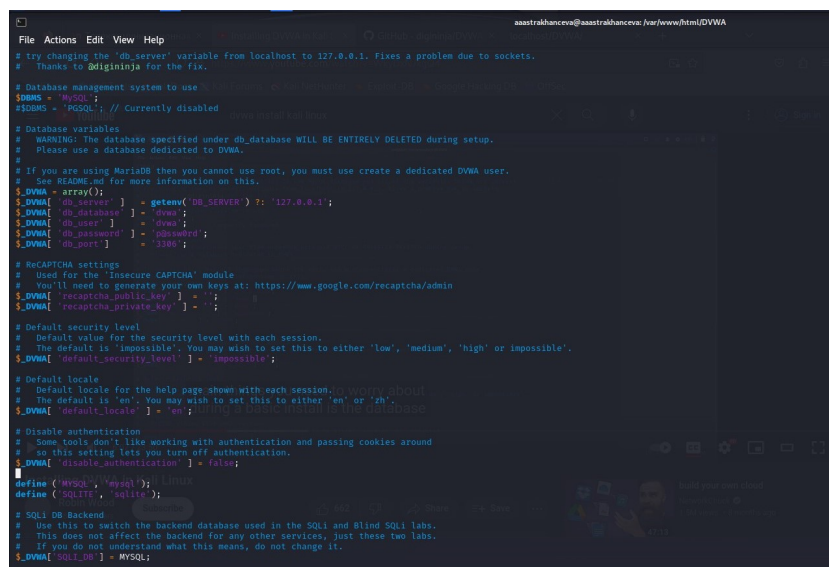
(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA/config]
$ cd ../..

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

(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA]
$
```

Рис. 3.6: Копирование конфигурационного файла

Далее просматриваем конфигурационный файл с помощью vim. Особое внимание уделяем паролю, в дальнейшем будем использовать его для создания базы данных (рис. 3.7).



```
File Actions Edit View Help
# Very 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 = 'PostgreSQL'; // 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' : 'localhost';
$DVWA[ 'db_database' ] = 'dvwa';
$DVWA[ 'db_user' ] = 'dvwa';
$DVWA[ 'db_password' ] = 'p0w3r!d1e';
$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'.
$DVWA[ 'default_security_level' ] = 'impossible';

# Default locale
# Default locale for the help page shown with each session. It defaults to 'en'.
# The default is 'en'. You may wish to set this to either 'en' or 'zh'.
$DVWA[ 'default_locale' ] = 'en';

# Disable authentication
# Some tools don't like working with authentication and passing cookies around
# so this setting lets you turn off authentication.
$DVWA[ 'disable_authentication' ] = false;

define( 'MYSQL', 'mysql' );
define( 'SQLITE', 'sqlite' );

# SQL DB Backend
# Use this to switch the backend database used in the SQLi and Blind SQLi labs.
# This does not affect the backend for any other services, just these two labs.
# If you do not understand what this means, do not change it.
$DVWA[ 'SQL_DB' ] = MYSQL;
```

Рис. 3.7: Просмотр конфигурационного файла

Запускаем mariadb для работы с базами данных (рис. 3.8).

```
(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA]
$ service mariadb start

(aaastrakhanceva@aaastrakhanceva)-[/var/www/html/DVWA]
$
```

Рис. 3.8: Запуск mariadb

Создаем новую базу данных. Для начала переходим в режим sudo-пользователя, потом с помощью команды `mysql` запускаем монитор MariaDB. Далее создаем новую базу данных, добавляем пользователя, указываем его пароль. (рис. 3.9).

```
root@aaastrakhanceva: ~
File Actions Edit View Help
(aaastrakhanceva@aaastrakhanceva)-[~]
$ sudo su -
[sudo] password for aaastrakhanceva:
(root@aaastrakhanceva)-[~]
# mysql
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

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 database dvwa;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> create user dvwa@localhost identified by 'p@ssw0rd';
Query OK, 0 rows affected (0.014 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.004 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.005 sec)

MariaDB [(none)]>
```

Рис. 3.9: Создание новой базы данных

На этом установка окончена, переходим на “<http://localhost/DVWA>” для дальнейших необходимых настроек. (рис. 3.10).

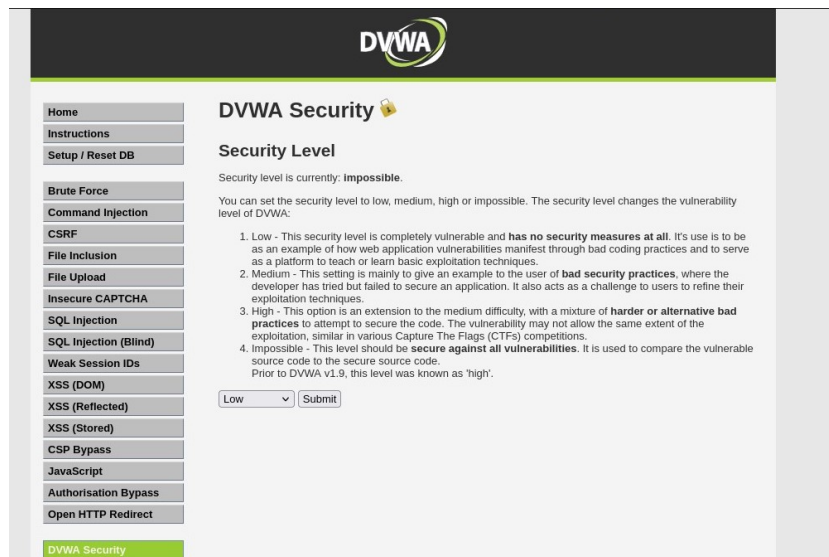


Рис. 3.10: Завершение установки

## 4 Выводы

В ходе выполнения второго этапа индивидуального проекта я ознакомилась с специально предназначенным для поиска уязвимостей веб приложением под названием Damn Vulnerable Web Application (DVWA).