

# **Индивидуальный проект. Этап 2**

**Установка DVWA**

Парфенова Елизавета Евгеньевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>4</b>	<b>Выводы</b>	<b>15</b>
	<b>Список литературы</b>	<b>16</b>

# Список иллюстраций

3.1	Клонирование репозитория DVWA . . . . .	8
3.2	Неудачная попытка входа . . . . .	9
3.3	Запуск сервера apache2 . . . . .	9
3.4	Удачное открытие страницы . . . . .	9
3.5	Неудачная попытка открытия DVWA . . . . .	10
3.6	Копирование конфигурационного файла . . . . .	10
3.7	Прочтение конфигурационного файла . . . . .	11
3.8	Открытие страницы <a href="http://localhost/DVWA/setup.php">http://localhost/DVWA/setup.php</a> . . . . .	12
3.9	Запуск базы данных . . . . .	12
3.10	Создание нового пользователя базы данных (dvwa) . . . . .	13
3.11	Проверка создания нового пользователя . . . . .	13
3.12	Данные входа на страницу . . . . .	14
3.13	Успешный вход на страницу DVWA . . . . .	14

## **Список таблиц**

# 1 Цель работы

Установка DVWA в гостевую систему к Kali Linux.

## 2 Теоретическое введение

Damn Vulnerable Web Application (DVWA) — это веб-приложение на PHP/MySQL, которое очень сильно уязвимо. Его главная цель — помочь профессионалам по безопасности протестировать их навыки и инструменты в легальном окружении, помочь веб-разработчикам лучше понять процесс безопасности веб-приложений и помочь и студентам и учителям в изучении безопасности веб-приложений в контролируемом окружении аудитории. [1]

Некоторые из уязвимостей веб приложений, который содержит DVWA:

- *Брутфорс*: Брутфорс HTTP формы страницы входа - используется для тестирования инструментов по атаке на пароль методом грубой силы и показывает небезопасность слабых паролей.
- *Исполнение (внедрение) команд*: Выполнение команд уровня операционной системы.
- *Межсайтовая подделка запроса (CSRF)*: Позволяет «атакующему» изменить пароль администратора приложений.
- *Внедрение (инклюд) файлов*: Позволяет «атакующему» присоединить удалённые/локальные файлы в веб приложение.
- *SQL внедрение*: Позволяет «атакующему» внедрить SQL выражения в HTTP из поля ввода, DVWA включает слепое и основанное на ошибке SQL внедрение.
- *Небезопасная выгрузка файлов*: Позволяет «атакующему» выгрузить вредоносные файлы на веб сервер.
- *Межсайтовый скриптинг (XSS)*: «Атакующий» может внедрить свои скрипты в веб приложение/базу данных. DVWA включает отражённую и хранимую

XSS.

- *Пасхальные яйца*: раскрытие полных путей, обход аутентификации и некоторые другие.

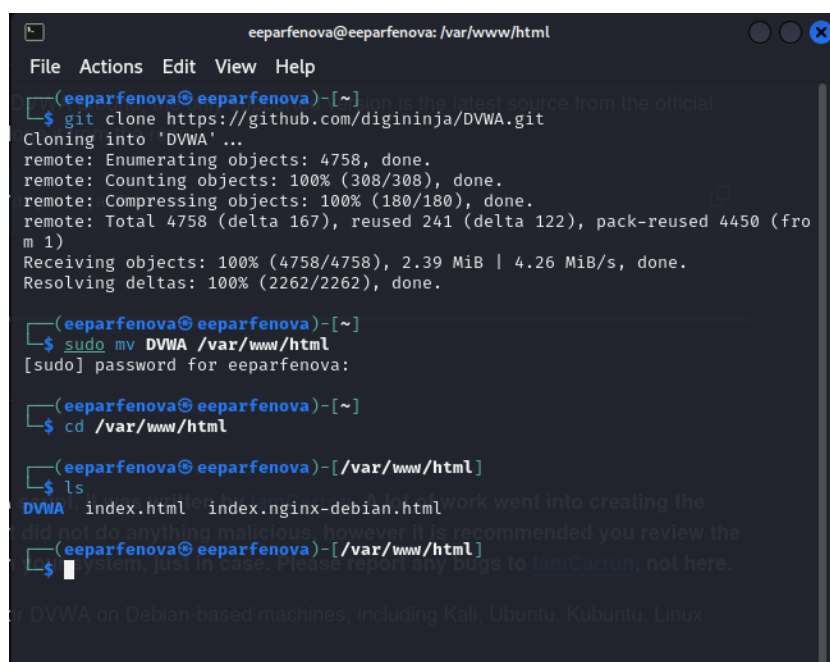
DVWA имеет несколько уровней безопасности, они меняют уровень безопасности каждого веб приложения в DVWA:

- *Невозможный* — этот уровень должен быть безопасным от всех уязвимостей. Он используется для сравнения уязвимого исходного кода с безопасным исходным кодом.
- *Высокий* — это расширение среднего уровня сложности, со смесью более сложных или альтернативных плохих практик в попытке обезопасить код. Уязвимости не позволяют такой простор эксплуатации как на других уровнях.
- *Средний* — этот уровень безопасности предназначен главным образом для того, чтобы дать пользователю пример плохих практик безопасности, где разработчик попытался сделать приложение безопасным, но потерпел неудачу.
- *Низкий* — этот уровень безопасности совершенно уязвим и совсем не имеет защиты. Его предназначение быть примером среди уязвимых веб приложений, примером плохих практик программирования и служить платформой обучения базовым техникам эксплуатации.

### 3 Выполнение лабораторной работы

Установку DVWA будем выполнять согласно инструкции в репозитории гитхаба (в том числе видео-инструкции) [2]

Начнем установку с клонирования указанного репозитория на наш компьютер. Для этого в терминале введём команду `git clone https://github.com/digininja/DVWA.git`. Далее по рекомендации перенесем созданный каталог в директорию `/var/www/html` командой `sudo mv DVWA /var/www/html` (зайдем с правами администратора) и проверим, что все получилось корректно. (рис. 3.1).



```
eeeparfenova@eeeparfenova: /var/www/html
File Actions Edit View Help
(eeparfenova@eeeparfenova)-[~]
$ git clone https://github.com/digininja/DVWA.git
Cloning into 'DVWA' ...
remote: Enumerating objects: 4758, done.
remote: Counting objects: 100% (308/308), done.
remote: Compressing objects: 100% (180/180), done.
remote: Total 4758 (delta 167), reused 241 (delta 122), pack-reused 4450 (from 1)
Receiving objects: 100% (4758/4758), 2.39 MiB | 4.26 MiB/s, done.
Resolving deltas: 100% (2262/2262), done.
(eeparfenova@eeeparfenova)-[~]
$ sudo mv DVWA /var/www/html
[sudo] password for eeeparfenova:
(eeparfenova@eeeparfenova)-[~]
$ cd /var/www/html
(eeparfenova@eeeparfenova)-[/var/www/html]
$ ls
DVWA  index.html  index.nginx-debian.html
(eeparfenova@eeeparfenova)-[/var/www/html]
$
```

Рис. 3.1: Клонирование репозитория DVWA

Далее попробуем зайти на сервер по ссылке `http://localhost`, но увидим, что попытка не удалась. (рис. 3.2)



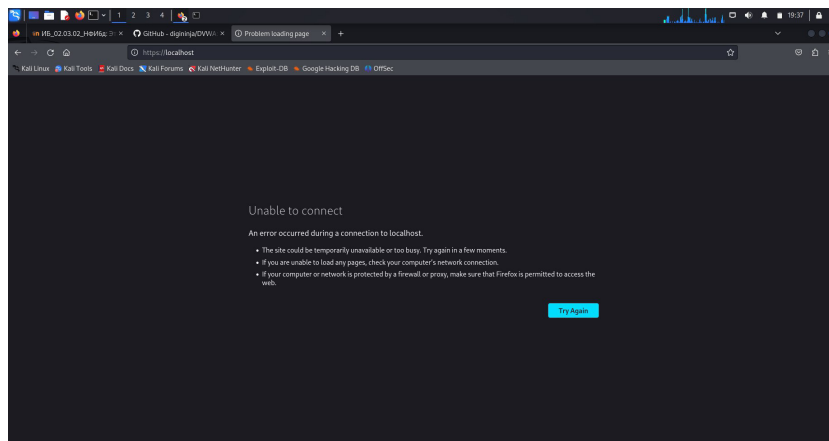


Рис. 3.2: Неудачная попытка входа

Запустим стандартный сервер apache2 командой *sudo service apache2 start*, (рис. 3.3) и попробовав перезагрузить страницу, увидим, что все сработало и открылась стартовая страница сервера apache2. (рис. 3.4)

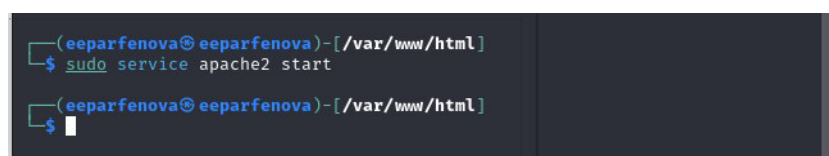


Рис. 3.3: Запуск сервера apache2

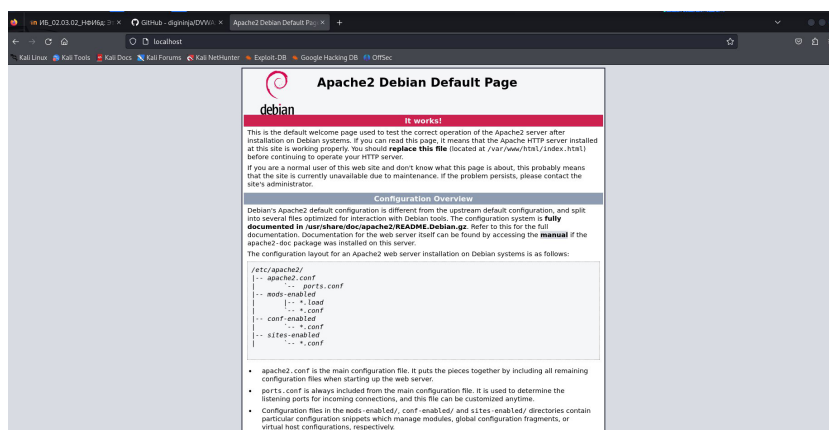


Рис. 3.4: Удачное открытие страницы

Далее попробуем перейти на страницу <http://localhost/DVWA/> (рис. 3.5), но увидим, что это вышло не совсем корректно. Как и требует текст высветившейся

ошибки, поработаем с файлом `config`.

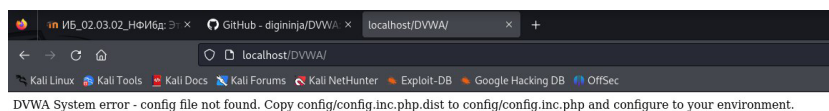


Рис. 3.5: Неудачная попытка открытия DVWA

Для этого перейдем в директорию DVWA и посмотрим, что лежит в директории `config` командой `ls`. Далее скопируем дист-версию конфигурационного файла в файл `config.inc.php` (для большей безопасности именно скопируем, а не перенесем) (рис. 3.6), а затем прочитаем файл командой `cat config/config.inc.php` (рис. 3.7). Из этого файла нам особенно понадобятся учетные данные для базы данных, которые мы не изменяем.

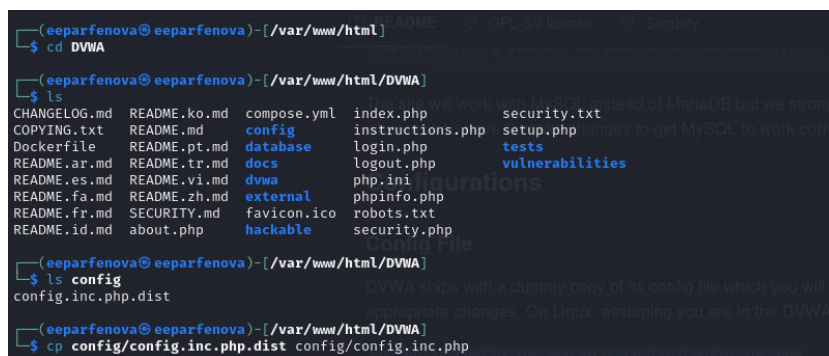


Рис. 3.6: Копирование конфигурационного файла

```
(eeparfenova@eeparfenova)-[/var/www/html/DVWA] cat config/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 DELETE
D during setup.
# Please use a database dedicated to DVWA.
#
# If you are using MariaDB then you cannot use root, you must use create a de
dicated 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' ] = 'dvwa';
$_DVWA[ 'db_password' ] = 'p@ssw0rd';
$_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/recaptch
a/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', 'm
edium', 'high' or 'impossible'.
$_DVWA[ 'default_security_level' ] = 'impossible';

# Default locale
# Default locale for the help page shown with each session, to the following by default.
# 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 are
und
# so this setting lets you turn off authentication.
$_DVWA[ 'disable_authentication' ] = false;
```

Рис. 3.7: Прочтение конфигурационного файла

Увидим, что мы смогли войти на страницу <http://localhost/DVWA/setup.php> (рис. 3.8), но при попытке создать базу данных мы ничего не получаем

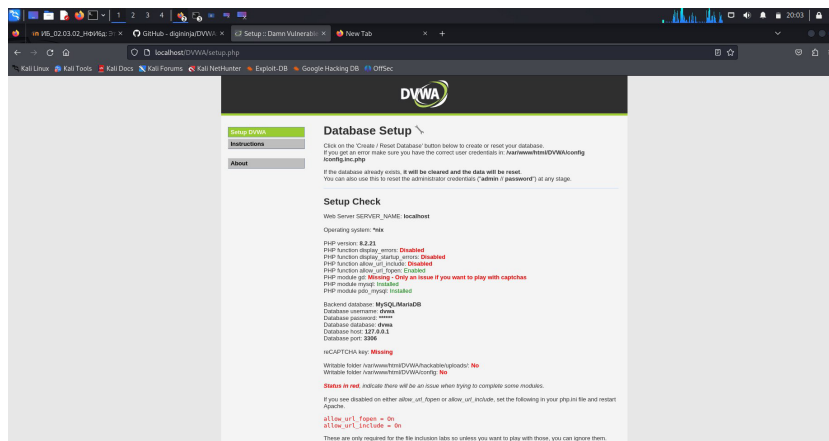


Рис. 3.8: Открытие страницы <http://localhost/DVWA/setup.php>

Запустим стандартную базу данных mariadb командой `service mariadb start` (рис. 3.9)

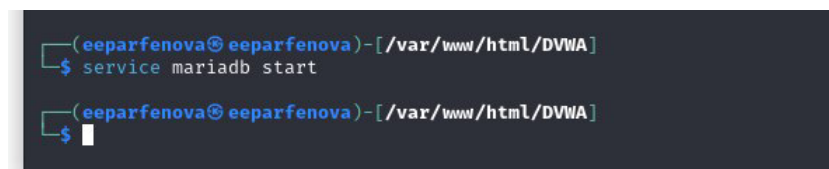


Рис. 3.9: Запуск базы данных

Далее в другом окне терминала, подключившись к БД от имени пользователя root командами `sudo -su` и `mysql`, создадим нового пользователя следующими команды (рис. 3.10), проверяя, чтобы вывод после них не свидетельствовал о какой-либо ошибке:

- `MariaDB [(none)]> create database dvwa;`
- `MariaDB [(none)]> create user dvwa@localhost identified by 'p@sswOrd';`
- `MariaDB [(none)]> grant all on dvwa. to dvwa@localhost;*`
- `MariaDB [(none)]> flush privileges;`

При введении первых трех команд ориентируемся на данные из конфигурационного файла, а последней командой мы перезапускаем БД.

```
(eeparfenova@eeparfenova)-[~]
$ sudo su -
(eeparfenova@eeparfenova)-[~]
$ mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 11.4.2-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)]> create database dvwa
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> create user dvwa@localhost identified by 'p@ssw0rd';
Query OK, 0 rows affected (0.016 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.007 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

Рис. 3.10: Создание нового пользователя базы данных (dvwa)

Проверим корректность создания нового пользователя bd с помощью команды *mysql -u dvwa -pp@ssw0rd*, открыв новое окно терминала. (рис. 3.11) Этой командой мы входим в базу данных по “учетной записи” созданного пользователя. Все получается корректно.

```
(eeparfenova@eeparfenova)-[~]
$ mysql -u dvwa -pp@ssw0rd
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 11.4.2-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)]>
```

Рис. 3.11: Проверка создания нового пользователя

После выполненных манипуляций попробуем снова создать базу данных через сайт. Для этого проскролим сайт вниз и нажмем на кнопку “Create/Reset Database”. Нас перекидывает на страницу входа, значит все получилось верно. Вводим в пустые поля стандартные “admin” и “password” (рис. 3.12) и попадаем на страницу DVWA (рис. 3.13).

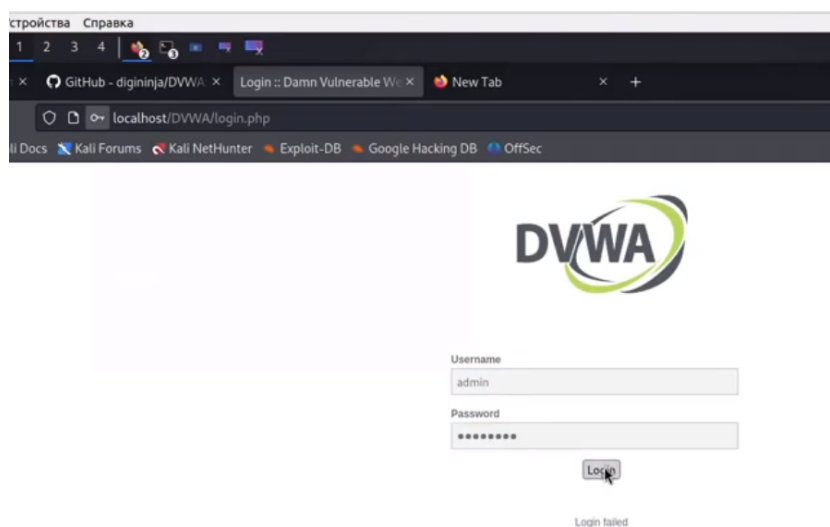


Рис. 3.12: Данные входа на страницу

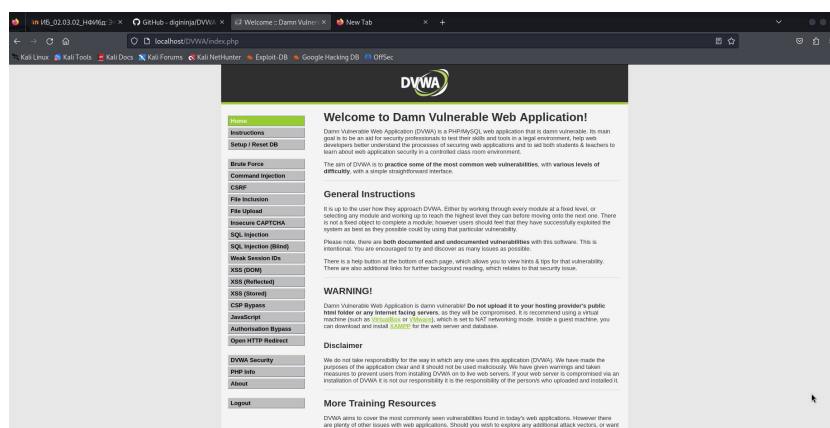


Рис. 3.13: Успешный вход на страницу DVWA

На этом стандартная установка DVWA окночена.

## 4 Выводы

Мы установили DVWA в гостевую систему к Kali Linux.

## Список литературы

1. Damn Vulnerable Web Application (DVWA) [Электронный ресурс]. Инструменты Kali Linux, 2024. URL: <https://kali.tools/?p=1820>.
2. DVWA [Электронный ресурс]. GitHub, Inc., 2024. URL: <https://github.com/digininja/DVWA>.