

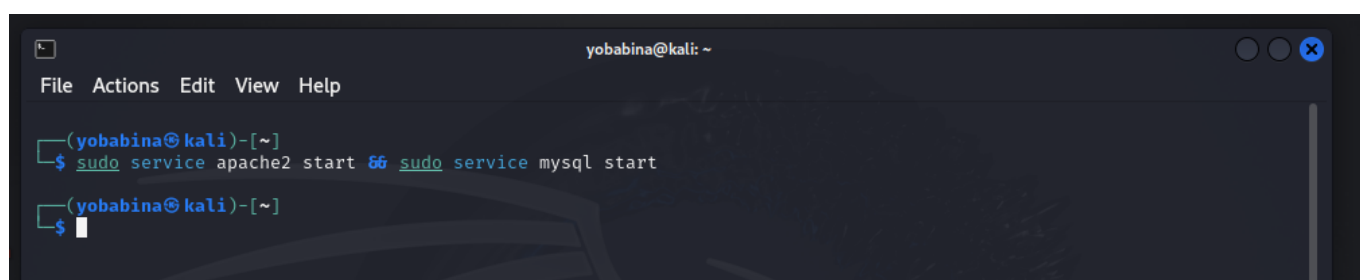
Презентация к 4 этапу индивидуального проекта

Цель работы

Приобретение практических навыков по использованию nikto - базового сканера безопасности веб-сервера.

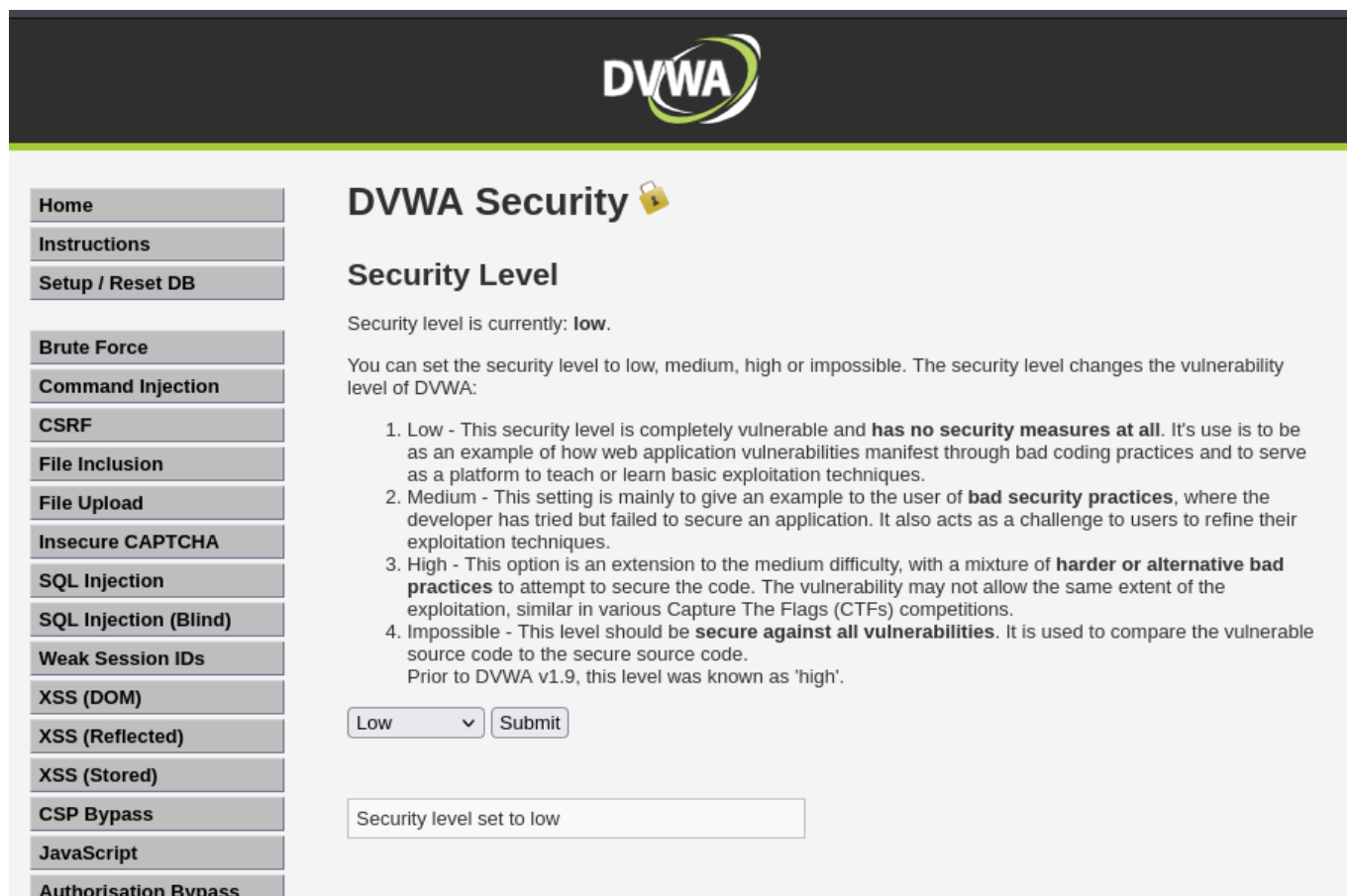
Выполнение работы

Запуск сервисов



```
yobabina@kali: ~  
File Actions Edit View Help  
(yobabina@kali)-[~]  
$ sudo service apache2 start && sudo service mysql start  
(yobabina@kali)-[~]  
$
```

Изменение настроек безопасности



The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The left sidebar contains a list of vulnerability categories: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, and Authorisation Bypass. The main content area is titled "DVWA Security" with a lock icon. Below the title, it states "Security level is currently: low." and provides instructions on how to set the security level to low, medium, high, or impossible. A list of four security levels is provided, each with a brief description. At the bottom, there is a dropdown menu set to "Low" and a "Submit" button. Below the button, a message box states "Security level set to low".

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

Утилита nikto

```
File Actions Edit View Help
(yobabina@kali)~[~]
$ nikto -h http://localhost/DVWA/
- Nikto v2.5.0

+ Target IP: 127.0.0.1
+ Target Hostname: localhost
+ Target Port: 80
+ Start Time: 2024-10-01 22:21:44 (GMT3)

+ Server: Apache/2.4.62 (Debian)
+ /DVWA/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /DVWA/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Root page /DVWA redirects to: login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ /DVWA/config/: Directory indexing found. 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.
+ /DVWA/tests/: This might be interesting. 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.
+ /DVWA/database/: Directory indexing found. 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.
+ /DVWA/docs/: Directory indexing found. This level should be secure against all vulnerabilities. It is used to compare the vulnerable source code to the secure source code.
+ /DVWA/login.php: Admin login page/section found.
+ /DVWA/.git/index: Git Index file may contain directory listing information.
+ /DVWA/.git/HEAD: Git HEAD file found. Full repo details may be present.
+ /DVWA/.git/config: Git config file found. Infos about repo details may be present.
+ /DVWA/.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ /DVWA/.dockerignore: .dockerignore file found. It may be possible to grasp the directory structure and learn more about the site.
+ 7850 requests: 0 error(s) and 16 item(s) reported on remote host
+ End Time: 2024-10-01 22:22:10 (GMT3) (26 seconds)

+ 1 host(s) tested
```

Повторное изменение настроек безопасности

Home

Instructions

Setup / Reset DB

Brute Force

Command Injection

CSRF

File Inclusion

File Upload

Insecure CAPTCHA

SQL Injection

SQL Injection (Blind)

Weak Session IDs

XSS (DOM)


XSS (Reflected)

XSS (Stored)

CSP Bypass

JavaScript

Authorisation Bypass



DVWA Security

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:

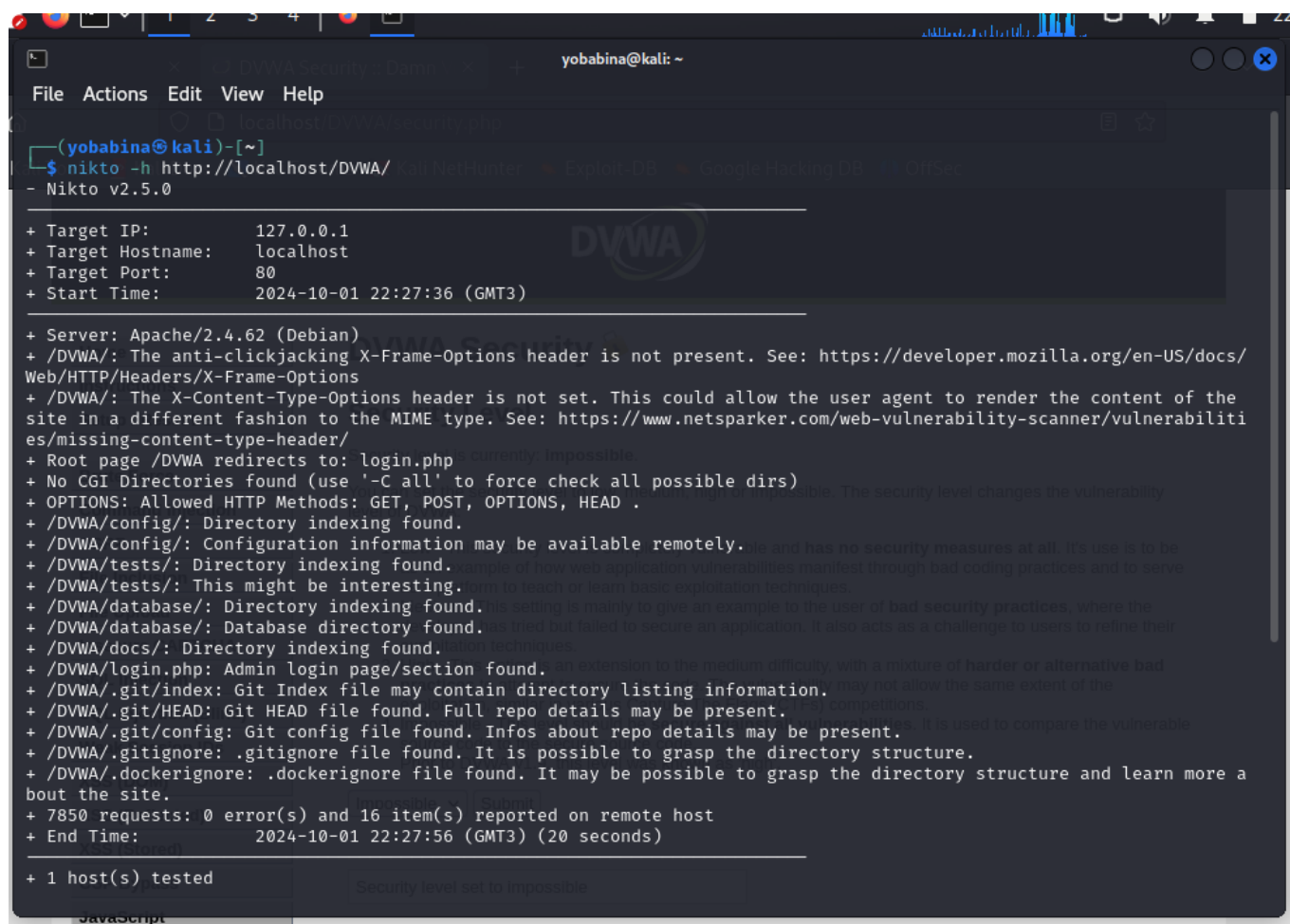
1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting 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'.

Impossible

Submit

Security level set to impossible

Повторный запуск утилиты



```
(yobabina@kali)-[~]
$ nikto -h http://localhost/DVWA/
- Nikto v2.5.0

+ Target IP: 127.0.0.1
+ Target Hostname: localhost
+ Target Port: 80
+ Start Time: 2024-10-01 22:27:36 (GMT3)

+ Server: Apache/2.4.62 (Debian)
+ /DVWA/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /DVWA/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Root page /DVWA redirects to: login.php - currently impossible
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ /DVWA/config/: Directory indexing found.
+ /DVWA/config/: Configuration information may be available remotely.
+ /DVWA/tests/: Directory indexing found.
+ /DVWA/tests/: This might be interesting.
+ /DVWA/database/: Directory indexing found.
+ /DVWA/database/: Database directory found.
+ /DVWA/docs/: Directory indexing found.
+ /DVWA/login.php: Admin login page/section found.
+ /DVWA/.git/index: Git Index file may contain directory listing information.
+ /DVWA/.git/HEAD: Git HEAD file found. Full repo details may be present.
+ /DVWA/.git/config: Git config file found. Infos about repo details may be present.
+ /DVWA/.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ /DVWA/.dockerignore: .dockerignore file found. It may be possible to grasp the directory structure and learn more about the site.
+ 7850 requests: 0 error(s) and 16 item(s) reported on remote host
+ End Time: 2024-10-01 22:27:56 (GMT3) (20 seconds)

+ 1 host(s) tested
```

Вывод

В рамках выполнения данной лабораторной работы я приобрела практический навык по использованию nikto - базового сканера безопасности веб-сервера.