# A Novel Web Application security vulnerability scanning tool.

MSc Cybersecurity
Industry Internship

Abhay Singh
Student ID: X21212341

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

**National College of Ireland**

**MSc Project Submission Sheet**

**School of Computing**

**Student Name:** Abhay Sureshkumar Singh

**Student ID:** X21212341

**Programme:** Masters in Cybersecurity **Year:** 2022-2023

**Module:** Industry Internship

**Lecturer:** Vikas Sahni

**Submission Due Date:** 04-09-2023

**Project Title:** A novel Web Application security vulnerability scanning tool

**Word Count:** 1188 **Page Count:** 12

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Abhay Singh

**Date:** 04-09-2023

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies) | □ |
| **Attach a Moodle submission receipt of the online project submission,** to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Configuration Manual

# A Novel Web Application security vulnerability scanning tool.

Abhay Singh
Student ID: X21212341

## 1 Introduction

For implementation of this project, please follow this configuration guide for the cutting-edge Web Application security vulnerability assessment tool "FlaskVulnerabilityScanner," which is based on the dependable Flask framework. The detailed instructions in this manual can be used to fine-tune parameters, guarantee flawless integration, and make the most of “FlaskVulnerabilityScanner”. This step-by-step methodology equips administrators and developers to efficiently strengthen their online applications, from basic setup to sophisticated modification. “FlaskVulnerabilityScanner” will act as your online assets' protector against cyber-attacks by utilizing Flask's agility.

## 2 Configuration

**Hardware Requirements:**

Processor: Dual-core processor (2 GHz or higher)
RAM: 2 GB
Storage: 10 GB of free disk space
Network: Internet connection for vulnerability database updates
Oracle-Virtual Box: Kali-Linux-2022.3-virtualbox-amd64[1]
Operating System: Linux OS (Debian 64-bit)

**Software Requirements:**

Python: version 3.0[2]
Redis Server: version 5:7.0.12-1[3]
Nessus Essential Community Edition: version 10.5.4
Skipfish: version 2.10b

[1] https://www.kali.org/docs/installation/hard-disk-install/
[2] https://wiki.python.org/moin/BeginnersGuide/Download
[3] https://redis.io/docs/getting-started/installation/install-redis-on-linux/

## Development Environment:

Integrated Development Environment (IDE): Visual Studio Code 1.81.1[4]
Text Editor: Any text editor of your choice (e.g., nano, vim, mousepad)

## Dependencies and Libraries:

Ensure that the following Python libraries are installed by running the command:

```
python3 -m pip install -r requirements.txt
```

Libraries listed in requirements.txt:

| |
|---|
| alabaster==0.7.12 |
| aniso8601==8.0.0 |
| Babel==2.8.0 |
| bcrypt==3.1.7 |
| beautifulsoup4==4.9.1 |
| certifi==2020.6.20 |
| cffi==1.14.0 |
| chardet==3.0.4 |
| click==7.1.2 |
| cryptography==3.0 |
| decorator==4.4.2 |
| dnspython==2.0.0 |
| docutils==0.16 |
| html5lib==1.1 |
| idna==2.10 |
| imagesize==1.2.0 |
| itsdangerous==1.1.0 |
| Jinja2==2.11.2 |
| MarkupSafe==1.1.1 |
| mysql-connector==2.2.9 |
| packaging==20.4 |
| paramiko==2.7.1 |
| Pillow==7.2.0 |
| psutil==5.7.2 |
| psycopg2-binary==2.8.5 |
| pycparser==2.20 |
| Pygments==2.6.1 |
| pymongo==3.11.0 |
| PyNaCl==1.4.0 |
| pyparsing==2.4.7 |
| PyPDF2==1.26.0 |
| python-nmap==0.6.1 |
| pytz==2020.1 |
| redis==3.5.3 |

[4] https://code.visualstudio.com/docs/setup/linux

| |
|---|
| reportlab==3.5.46 |
| requests==2.24.0 |
| simplejson==3.17.2 |
| six==1.15.0 |
| snowballstemmer==2.0.0 |
| soupsieve==2.0.1 |
| Sphinx==3.1.2 |
| sphinx-rtd-theme==0.5.0 |
| sphinxcontrib-applehelp==1.0.2 |
| sphinxcontrib-devhelp==1.0.2 |
| sphinxcontrib-htmlhelp==1.0.3 |
| sphinxcontrib-jsmath==1.0.1 |
| sphinxcontrib-qthelp==1.0.3 |
| sphinxcontrib-serializinghtml==1.1.4 |
| urllib3==1.25.9 |
| validators |
| webencodings==0.5.1 |
| Werkzeug==1.0.1 |
| flask_httpauth |
| flask_restful |
| flask |
| validators |
| requests |

It is highly recommended to use a Linux-based operating system for optimal performance and security. Make sure to use the updated system packages and dependencies.

# 3 Implementation

Step 1: Install the latest version of python and redis server using following command:

```
#Updating package installer and running the install command
sudo apt-get update
sudo apt-get install python

#Install redis server to run the web application (version (5:7.0.12-1))
sudo apt-get install redis
```

Step 2: Create a python virtual environment to avoid integration of other file projects making it difficult to run the web application.

```
#Create a python virtual environment named env (envnew is already created)
python3 -m venv env

#Activate the python environment
```

source env/bin/activate

```
#Create a python virtual environment named env (envnew is already created)
python3 -m venv env

#Activate the python environment
source env/bin/activate
```

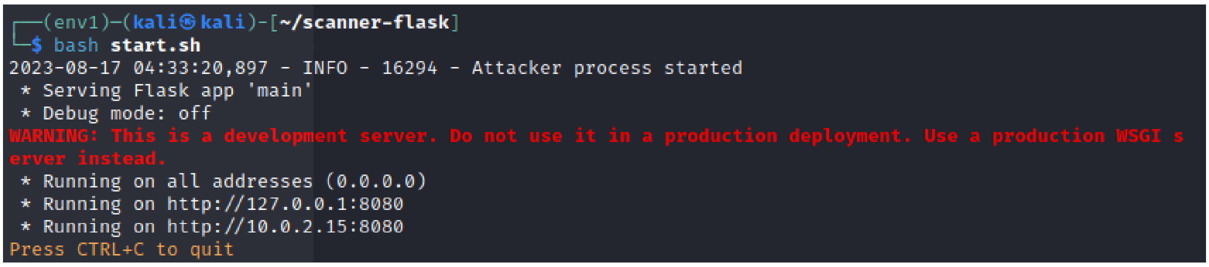
Step 3: Start the redis server from the main terminal or inside the python virtual environment.

```
sudo systemctl start redis-server
```

Step 4: Run the python library installable to download all the dependencies and finally run shell script file named start.sh to run the web application on the localhost.

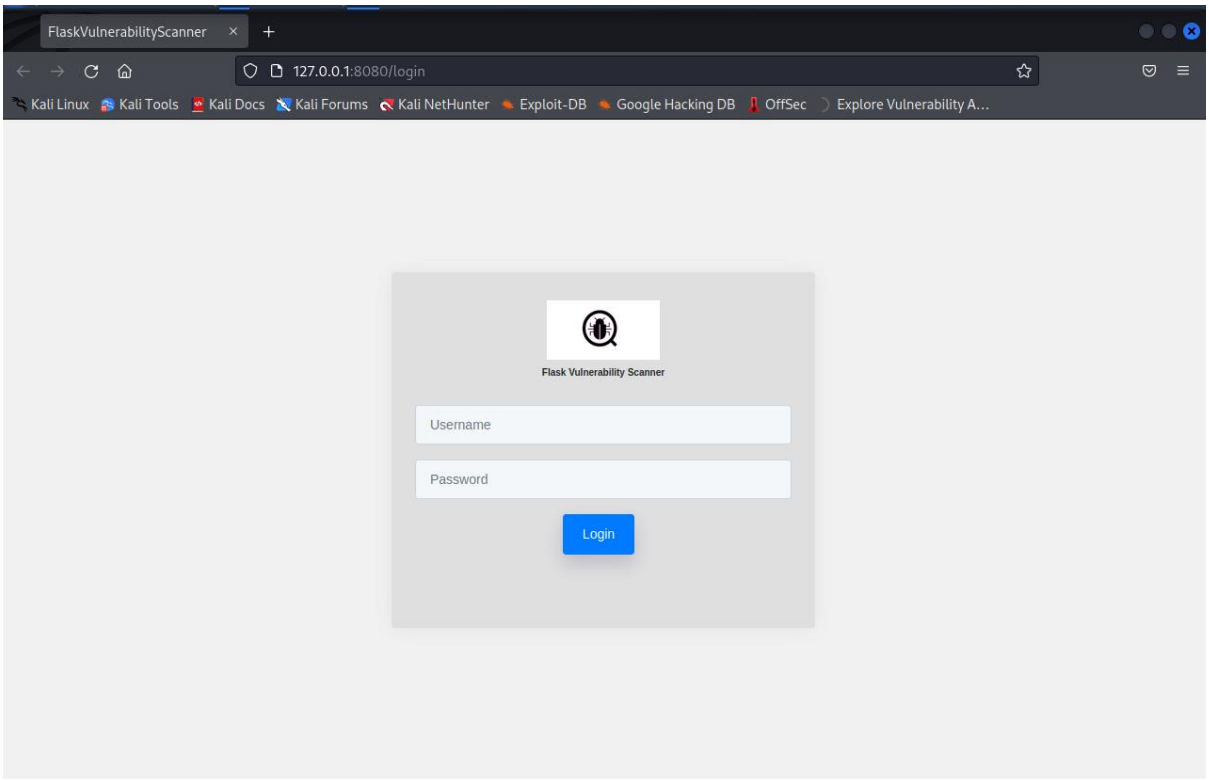
```
#Ensure that the following Python libraries are installed by running the command:
python3 -m pip install -r requirements.txt

#Use the bash shell or you can use any other preferred shell to run it.
bash start.sh
```



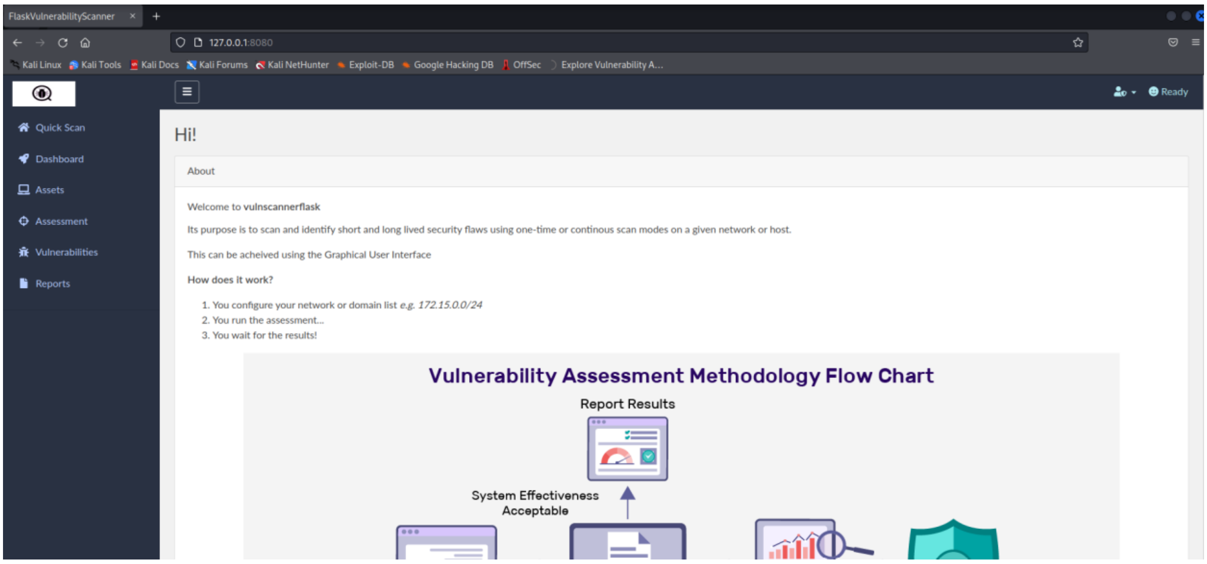
```
┌──(env1)-(kali㉿kali)-[~/scanner-flask]
└─$ bash start.sh
2023-08-17 04:33:20,897 - INFO - 16294 - Attacker process started
 * Serving Flask app 'main'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI s
erver instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8080
 * Running on http://10.0.2.15:8080
Press CTRL+C to quit
```

Step 5: Clicking on the addresses provided will redirect to the login Page of the application:



Enter the credentials as below:

username: admin
password: admin
Welcome Page: It’s directed to a welcome page of the app as below.



Hi!

About

Welcome to **vulnscannerflask**

Its purpose is to scan and identify short and long lived security flaws using one-time or continous scan modes on a given network or host.

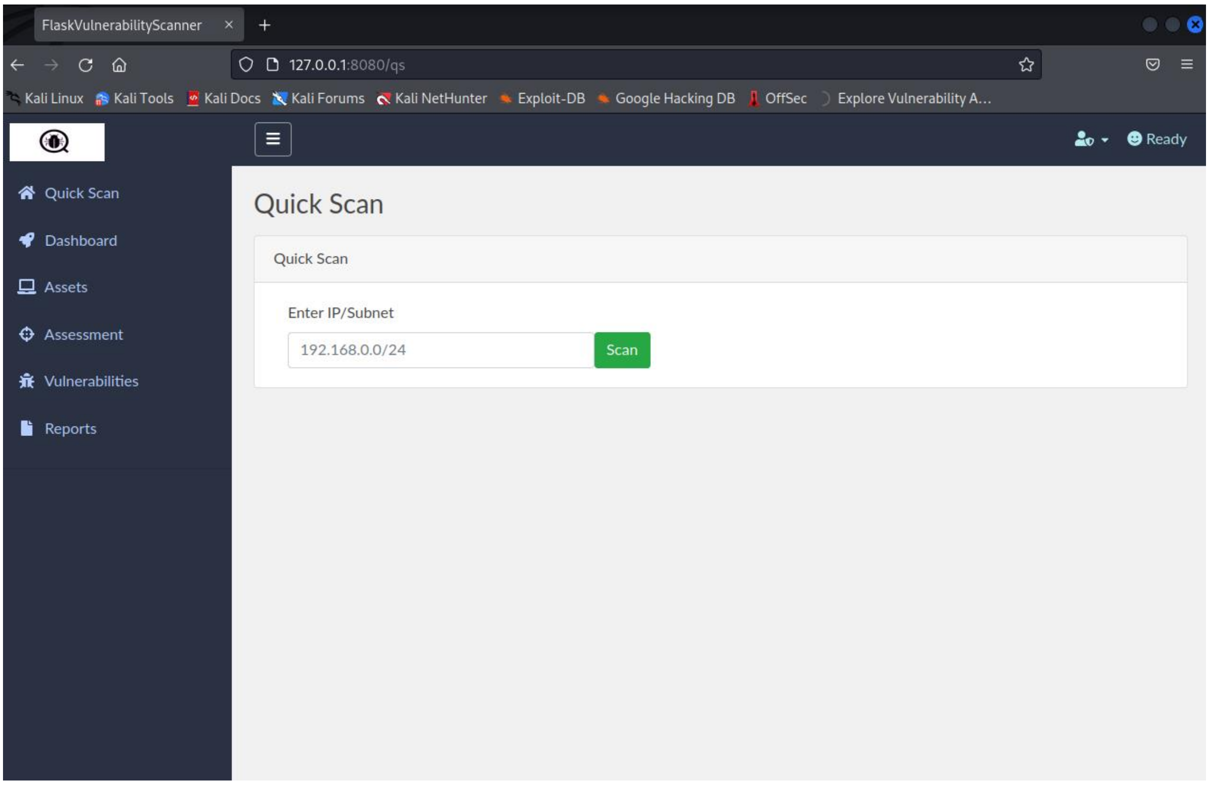
This can be acheived using the Graphical User Interface

**How does it work?**

1. You configure your network or domain list *e.g. 172.15.0.0/24*
2. You run the assessment...
3. You wait for the results!

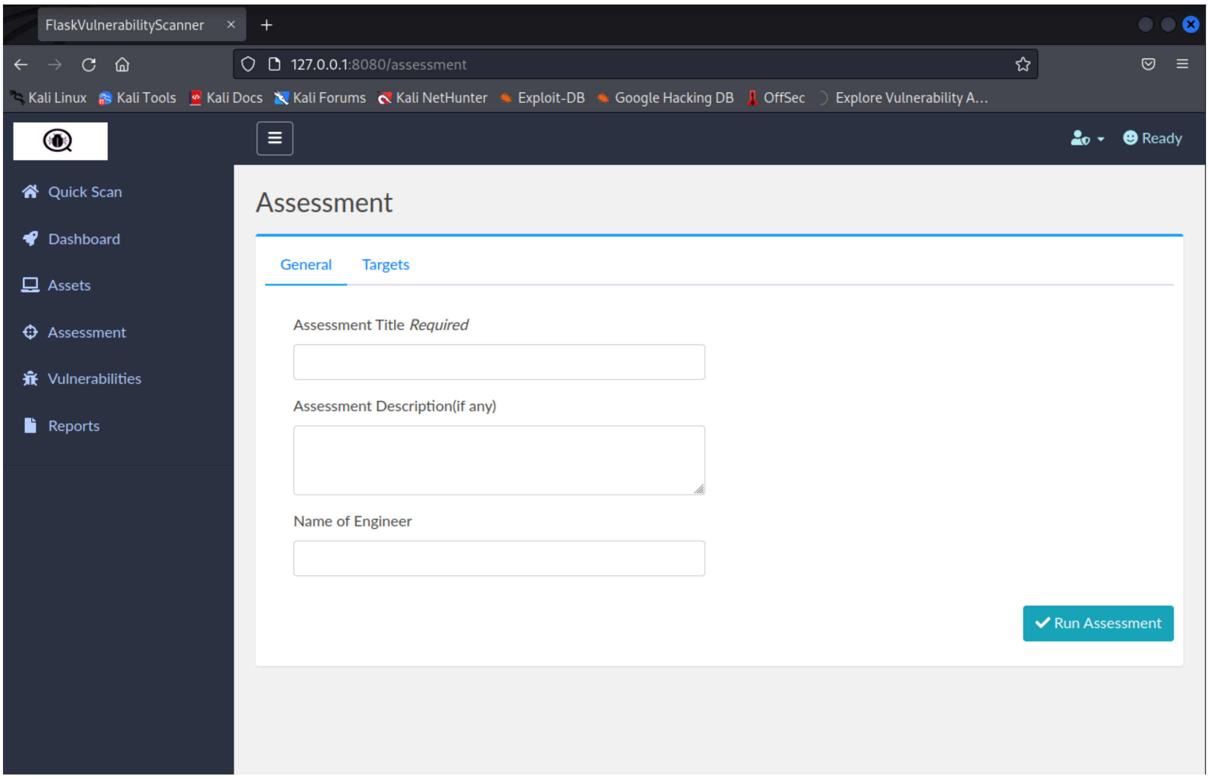
The assessment or scanning can be started by two methods:

Quick Scan: Enter the IP address or website domain that you wish to scan.

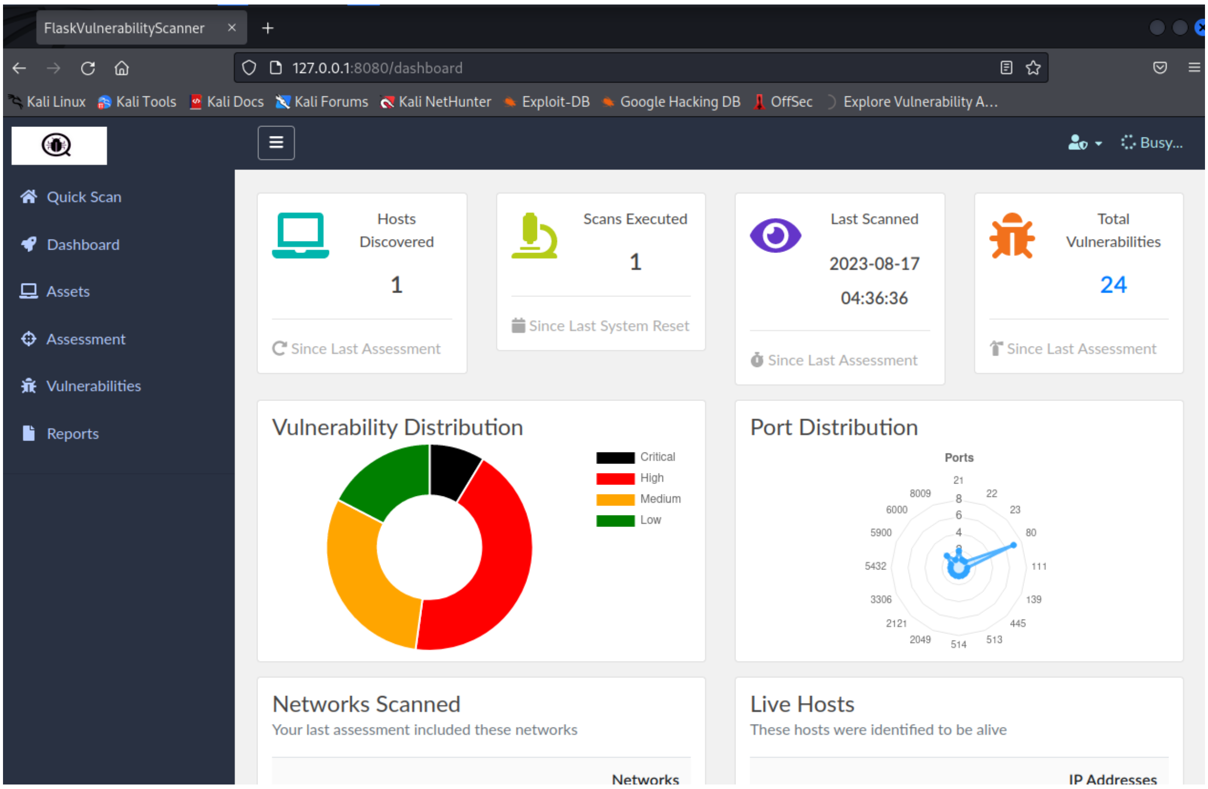


192.168.0.0/24

Assessment: The assessment would require inputs to provide few details of the scan to give a customized report consolidating those details. The General as well as Targets section needs to be filled to run the assessment.



Dashboard: After the scanning is successful, it will provide a dynamic output about the details of the scan on the dashboard page.



Hosts Discovered

1

Since Last Assessment

Scans Executed

1

Since Last System Reset

Last Scanned

2023-08-17

04:36:36

Since Last Assessment

Total Vulnerabilities

24

Since Last Assessment

Vulnerability Distribution

Port Distribution

Networks Scanned

Your last assessment included these networks

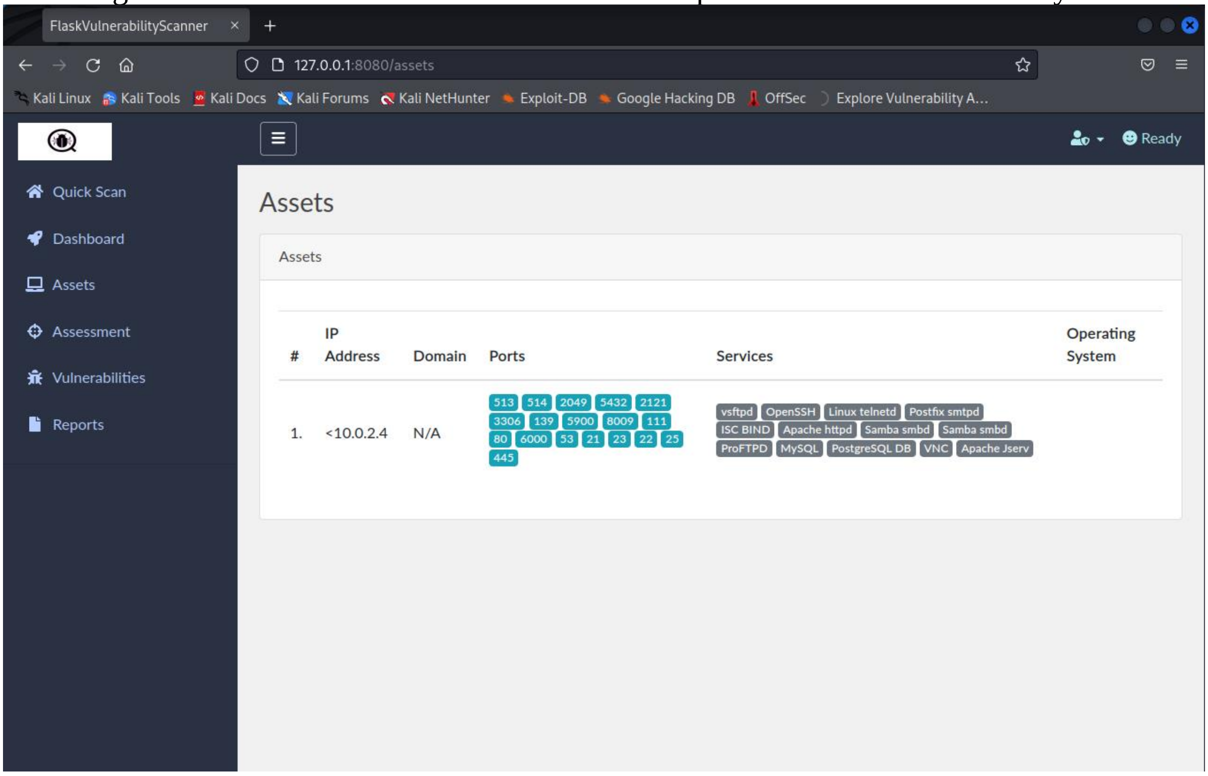
Networks

Live Hosts

These hosts were identified to be alive

IP Addresses

Assets: This is the output page where all the ports and devices that are scanned will be listed and along with it will mention the services that are responsible for the vulnerability.

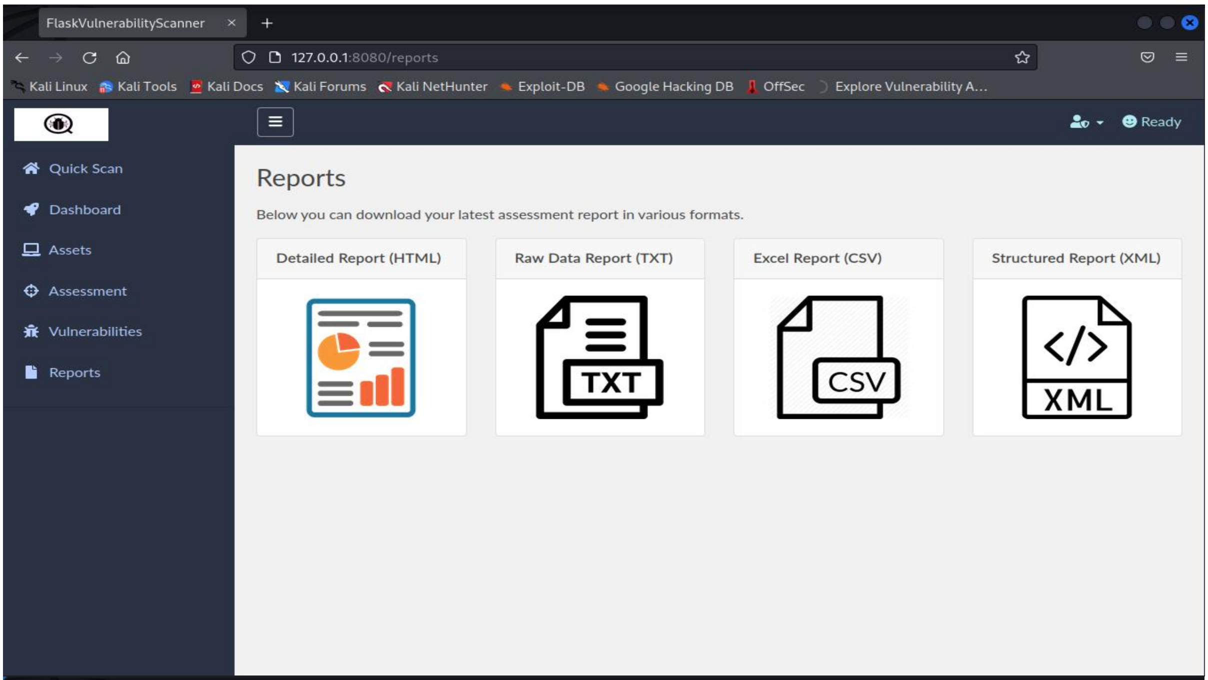


Assets

Assets

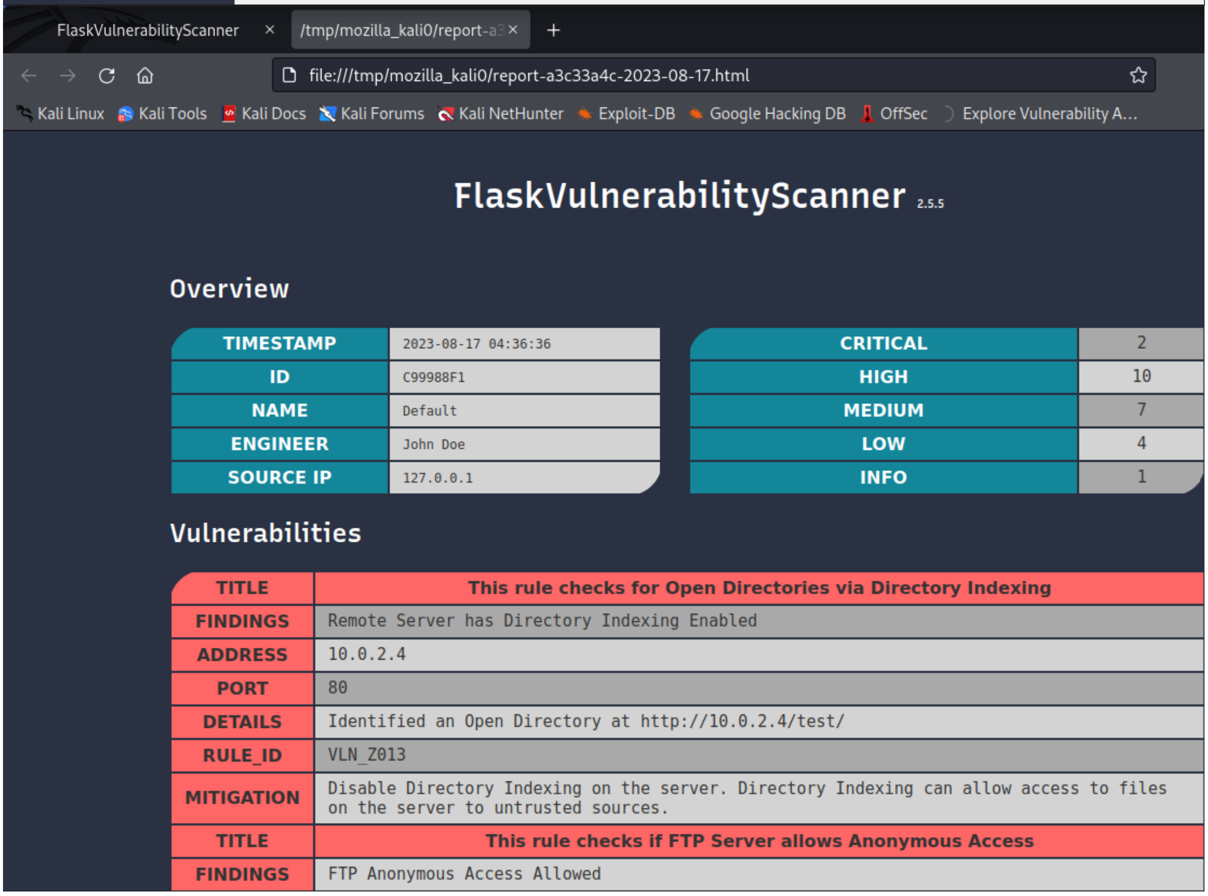
| # | IP Address | Domain | Ports | Services | Operating System |
|---|---|---|---|---|---|
| 1. | <10.0.2.4 | N/A | 513 514 2049 5432 2121 3306 139 5900 8009 111 80 6000 53 21 23 22 25 445 | vsftpd OpenSSH Linux telnetd Postfix smtpd ISC BIND Apache httpd Samba smbd Samba smbd ProFTPD MySQL PostgreSQL DB VNC Apache Jserv | |

Report: A documentation tab for all the vulnerabilities that need to be addressed is sent in form of a report in 4 types such as html, txt, xml, or csv. Any formats can be downloaded of it as per the requirement.



Reports

Below you can download your latest assessment report in various formats.


FlaskVulnerabilityScanner
/tmp/mozilla_kali0/report-a
file:///tmp/mozilla_kali0/report-a3c33a4c-2023-08-17.html
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Explore Vulnerability A...

FlaskVulnerabilityScanner 2.5.5

Overview

| | |
|---|---|
| TIMESTAMP | 2023-08-17 04:36:36 |
| ID | C99988F1 |
| NAME | Default |
| ENGINEER | John Doe |
| SOURCE IP | 127.0.0.1 |

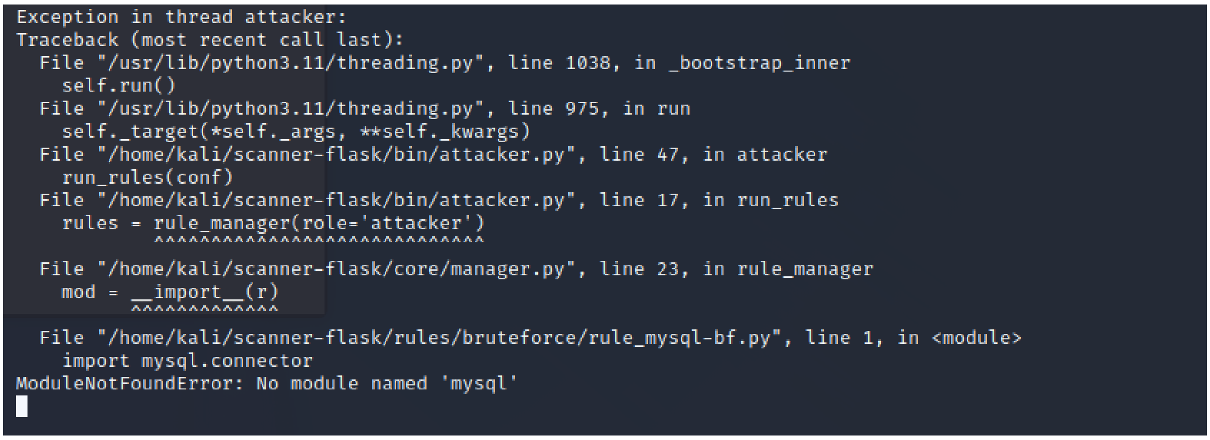
| | |
|---|---|
| CRITICAL | 2 |
| HIGH | 10 |
| MEDIUM | 7 |
| LOW | 4 |
| INFO | 1 |

Vulnerabilities

| | |
|---|---|
| TITLE | This rule checks for Open Directories via Directory Indexing |
| FINDINGS | Remote Server has Directory Indexing Enabled |
| ADDRESS | 10.0.2.4 |
| PORT | 80 |
| DETAILS | Identified an Open Directory at http://10.0.2.4/test/ |
| RULE_ID | VLN_Z013 |
| MITIGATION | Disable Directory Indexing on the server. Directory Indexing can allow access to files on the server to untrusted sources. |
| TITLE | This rule checks if FTP Server allows Anonymous Access |
| FINDINGS | FTP Anonymous Access Allowed |

Common Troubleshooting for errors while launching the project:

There were few errors that were encountered while launching this project due to a long list of python libraries and dependencies. An error might be caused because of installation issues of all the tools listed above as in the requirements.txt file like this:



```
Exception in thread attacker:
Traceback (most recent call last):
  File "/usr/lib/python3.11/threading.py", line 1038, in _bootstrap_inner
    self.run()
  File "/usr/lib/python3.11/threading.py", line 975, in run
    self._target(*self._args, **self._kwargs)
  File "/home/kali/scanner-flask/bin/attacker.py", line 47, in attacker
    run_rules(conf)
  File "/home/kali/scanner-flask/bin/attacker.py", line 17, in run_rules
    rules = rule_manager(role='attacker')
            ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/kali/scanner-flask/core/manager.py", line 23, in rule_manager
    mod = __import__(r)
          ^^^^^^^^^^^^^
  File "/home/kali/scanner-flask/rules/bruteforce/rule_mysql-bf.py", line 1, in <module>
    import mysql.connector
ModuleNotFoundError: No module named 'mysql'
```

For such issues, it needs to install the packaged library or tool using the below command manually inside the python virtual environment:

```
#General command
python3 -m pip install package name

#As in the example of the image above
python3 -m pip install mysql
```

# References

Team, F.S. (2010) Welcome to flask¶, Welcome to Flask - Flask Documentation (2.3.x). Available at: https://flask.palletsprojects.com/en/2.3.x/ (Accessed: 18 August 2023).

*The Python Standard Library* (2023a) *Python documentation*. Available at: https://docs.python.org/3/library/index.html (Accessed: 18 August 2023).

JonCyberGuy, J. (2023) *Joncyberguy/vulnerabilitymanagement: This is a walkthrough of how I created a virtual machine environment using vmware running Windows 10. I did this project to gain experience with Nessus Essentials and learn how to scan for vulnerabilities and remediate them. this project will showcase two of the main steps in the vulnerability management lifecycle. I will be using nessus essentials to scan local VMS hosted on vmware workstation in order run credentialed scans to discover vulnerabilities, remediate some of the vulnerabilities, then perform a rescan to verify remediation.*, *GitHub*. Available at: https://github.com/JonCyberGuy/VulnerabilityManagement (Accessed: 18 August 2023).

## 3.1 Appendix H – Monthly Internship Activity Report

The Internship Activity Report is a 1-page monthly summary of the activities performed by you and what you have learned during that month. The Internship Activity Report must be signed off by your Company and included in the configuration manual as part of the portfolio submission.

Student Name: Abhay Singh Student number: x21212341

Company: Knock Security Solutions Month Commencing:
June-August 2023

In June, I began gathering information regarding Web development frameworks and methodologies required to develop a web application. I referred to some research papers based on Python frameworks to figure out the simplest web framework. I also studied OWASP Top 10 Vulnerabilities for insights and created a reconnaissance checklist.
In first week of July, I started exploring different vulnerability scanner tools as requested by my manager. He provided me a list of GitHub applications that helped me to build my design structure for my proposed application.
In the second week of July, I started the documentation process and Literature survey.
In July third week, I worked on integrating the applications code to add all new features.
In fourth week, I went on to finalize the code structure and submitted the progress report to my manager. I got an approval on the reviewed papers and information from my team and my supervisor.
In August 1st week, I continued to explore different vulnerabilities other than the common vulnerabilities and went on to explore NVD/NIST websites.
In 2nd week of August, I finished the research methodology and Design part of the research paper.
In 3rd week of August, I referenced the Nessus and Skipfish tools to check the evaluation part and submitted the report for Manager's approval and final sign-off.

Employer comments

The Web Vulnerability Scanner developed by Abhay during his internship exemplified a committed and enthusiastic individual. His skill in incorporating research findings, particularly from IEEE papers, into the project demonstrated an in-depth comprehension of penetration testing methodologies. His professionalism and collaboration were highlighted by his proactive communication, meticulous documentation of vulnerabilities, and accurate security recommendations.

Student Signature: Abhay Singh Date: 24/08/2023

Industry Supervisor Signature: Date: 24/08/2023