MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

MINI PROJECT REPORT

Internet Technologies Lab (CSE 3262)

vvGvv

VIRUS VAULT

Sanjna Mallappa – 200905019
Rushin Reddy Ramesh – 200905066

Department of Computer Science and Engineering
Manipal Institute of Technology, Manipal.
April 2023

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Manipal
26/04/2023

CERTIFICATE

This is to certify that the project titled **vvGvv - VIRUS VAULT** is a record of the bonafide work done by **Sanjna Mallappa – 200905019 and Rushin Reddy Ramesh – 200905066** submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology (B.Tech.) in COMPUTER SCIENCE & ENGINEERING of Manipal Institute of Technology, Manipal, Karnataka, (A Constituent Institute of Manipal Academy of Higher Education), during the academic year 2022-2023.

Name and Signature of Examiners:

TABLE OF CONTENTS

ABSTRACT

1: INTRODUCTION

2: PROBLEM STATEMENT & OBJECTIVES

3: METHODOLOGY

4: RESULTS & SNAPSHOTS

5: CONCLUSION

6: LIMITATIONS & FUTURE WORK

7: REFERENCES

ABSTRACT

The abstract provides a brief overview of the entire report, summarizing the key points and highlighting the most significant results. In this report, we will discuss the development of a web application using Django, which allows users to upload files and scans them for malware using ClamAV. The report outlines the problem statement, objectives, methodology, results, and conclusion of the project.

INTRODUCTION

In this report, we will introduce a web application built with Django that provides a user-friendly interface for uploading files and scanning them for malware using ClamAV. Malware poses a significant threat to users' data and privacy, and thus there is a growing need for efficient and reliable malware scanning tools. Our project addresses this need by providing a simple and accessible web-based solution that utilizes the power of ClamAV to detect and remove malware from uploaded files. This report aims to provide a comprehensive overview of the project's objectives, methodology, results, and conclusion, and to evaluate the effectiveness and efficiency of our web application in detecting and removing malware.

PROBLEM STATEMENT & OBJECTIVES

The problem statement is to develop a web application that enables users to upload files and scan them for malware. The primary objectives of the project are:

- Develop a web application using Django that enables users to securely login and upload files.
- Integrate ClamAV antivirus engine to scan uploaded files for malware.
- Provide an easy-to-use interface for users to check if their files are infected with malware.

METHODOLOGY

This chapter outlines the methods used in this project to achieve the stated objectives. The methodology used in this project is divided into two main parts: the web application development methodology and the malware detection methodology.

3.1 Web Application Development Methodology

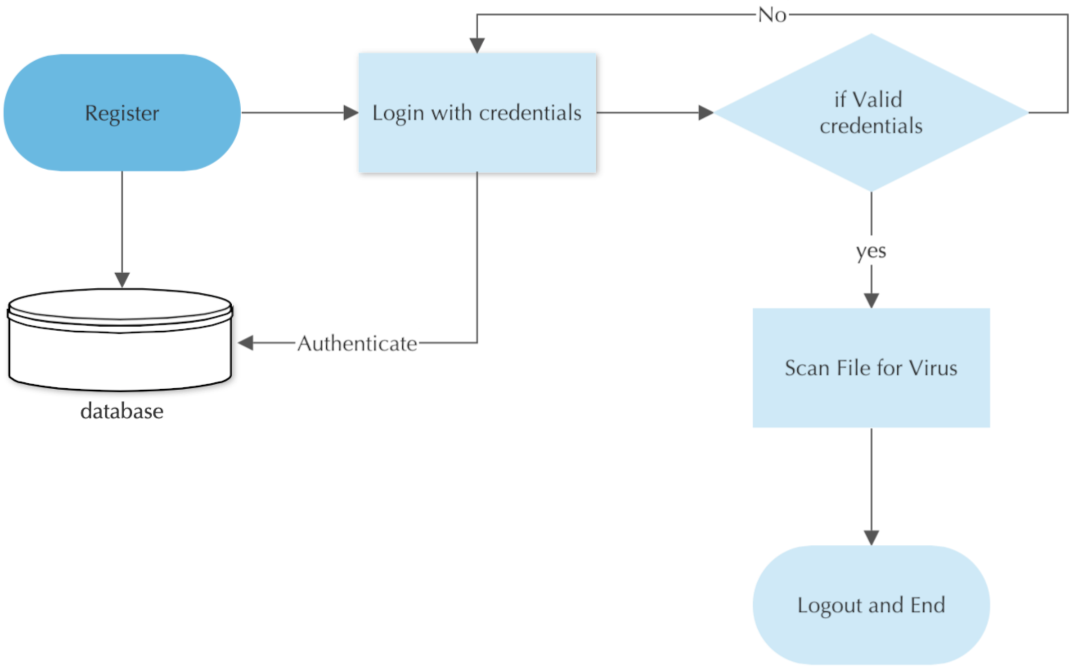
For the development of the web application, the Agile software development methodology was used. Agile development is an iterative and incremental approach to software development, where requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams. The Agile methodology was chosen for its flexibility, which allowed for changes to be made to the project requirements and scope as the project progressed.

3.2 Malware Detection Methodology

The malware detection methodology used in this project is based on the ClamAV antivirus engine. ClamAV is an open-source antivirus engine designed to detect malicious software such as viruses, trojans, and malware. ClamAV is widely used in the industry due to its high accuracy and low false positive rates.
The malware detection process involves scanning uploaded files for malware using the ClamAV antivirus engine. The file is first temporarily stored on the server and then passed to the ClamAV engine for scanning. The ClamAV engine returns a status report indicating whether the file is clean or infected with malware. If malware is detected, the type of malware is identified and reported back to the user.

In order to ensure the ClamAV engine is always up-to-date with the latest malware signatures, the engine is set up to automatically update itself on a daily basis. This ensures that the malware detection capabilities of the engine are always current and up-to-date with the latest malware threats.
The malware detection methodology was implemented using the pyclamd Python library, which provides a simple interface for communicating with the ClamAV antivirus engine over Unix sockets. The library was integrated into the Django web application to provide seamless malware detection capabilities for uploaded files.

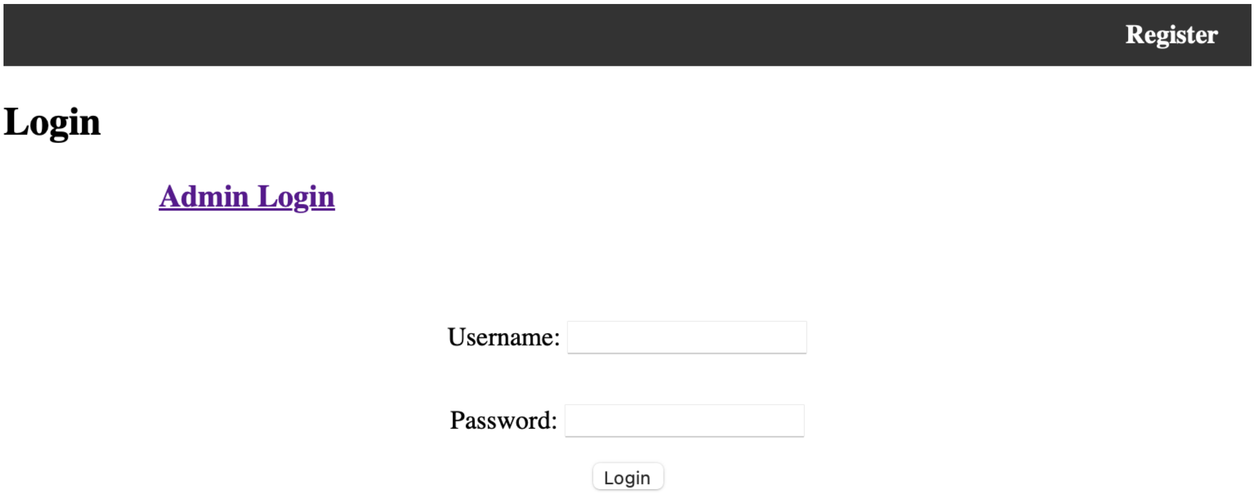
Flowchart:



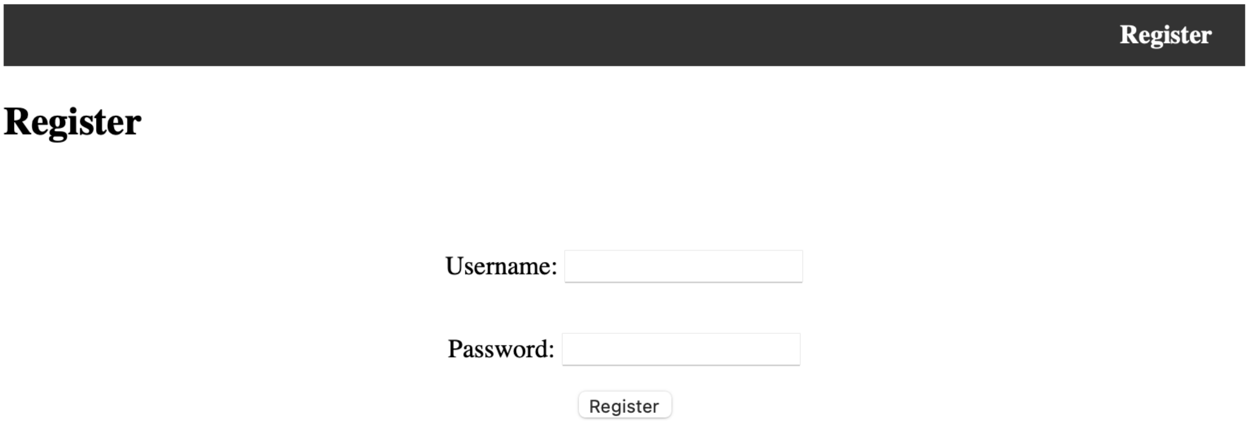
RESULTS & SNAPSHOTS

The web application was successfully developed using Django, and ClamAV was integrated to scan uploaded files for malware. The application was tested, and the results were satisfactory. The application provides an easy-to-use interface for users to upload files and check if they are infected with malware.

Login Page:



Register Page:

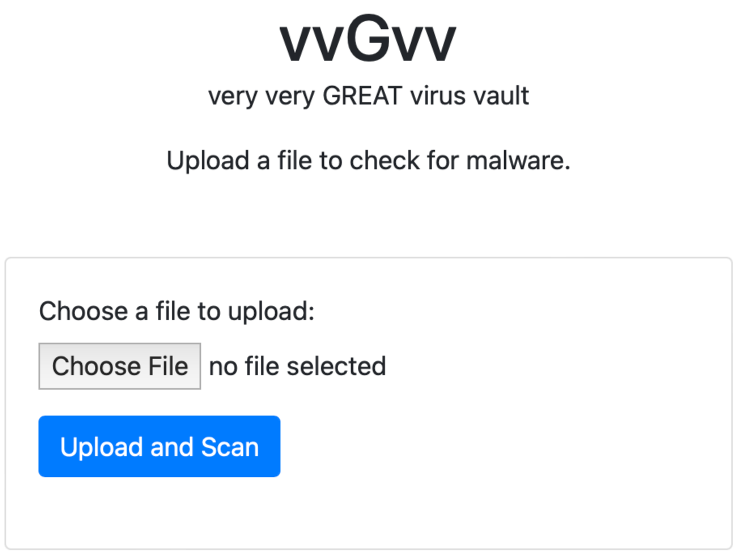


Register

Username:

Password:

Scan Page:



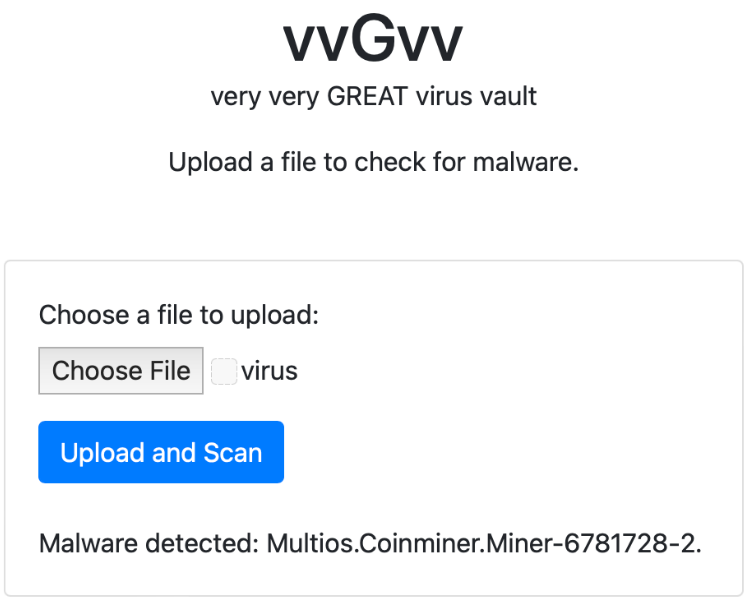
vvGvv

very very GREAT virus vault

Upload a file to check for malware.

Choose a file to upload:

Virus Scanned:


vvGvv

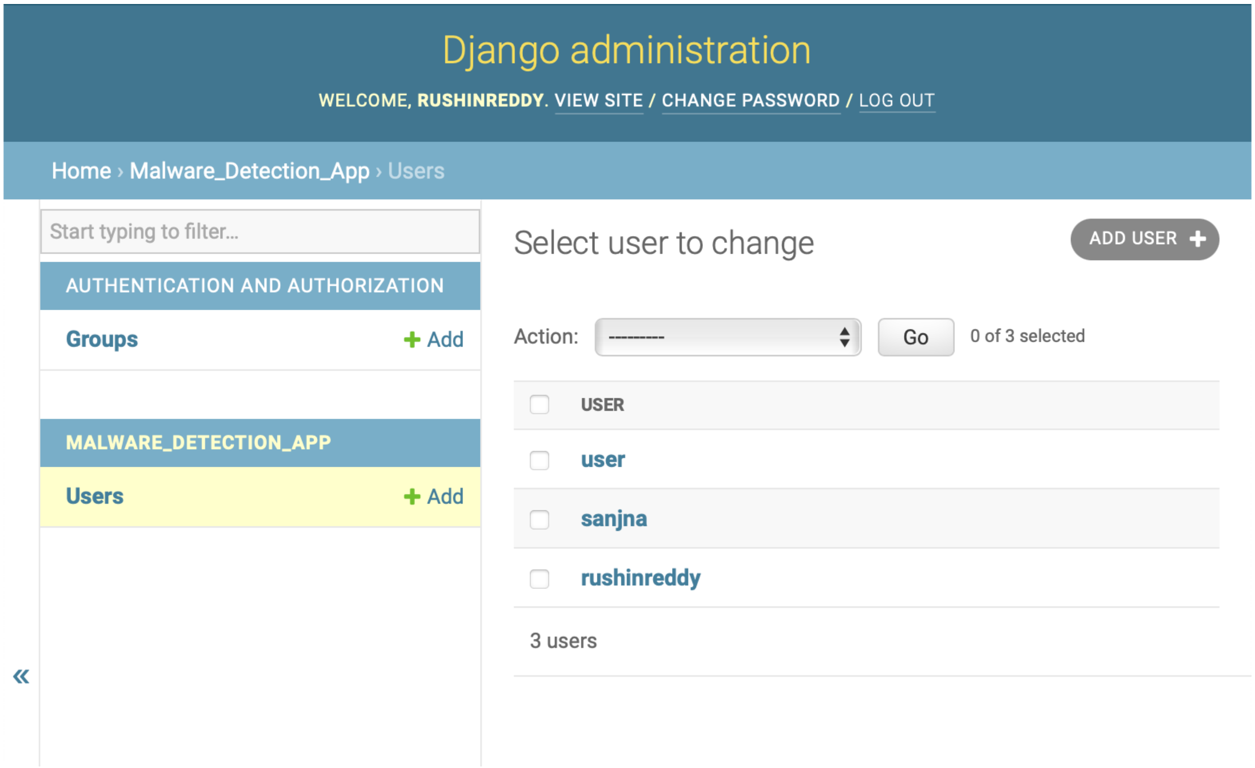
very very GREAT virus vault

Upload a file to check for malware.

Choose a file to upload:

Malware detected: Multios.Coinminer.Miner-6781728-2.

Admin Page:


Django administration

WELCOME, RUSHINREDDY. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Malware_Detection_App › Users

Select user to change

Action: --------- 0 of 3 selected

□	USER
□	user
□	sanjna
□	rushinreddy

3 users

CONCLUSION

In conclusion, the project was successful in developing a web application that enables users to upload files and scan them for malware using the ClamAV antivirus engine. The application provides an easy-to-use interface for users to check if their files are infected with malware. The project can be extended to include additional features such as email notifications and scheduled scans.

LIMITATIONS AND FUTURE WORK

Limitations:

- The current implementation only supports scanning files using ClamAV, which may not be sufficient for all types of malware detection.
- The current implementation does not provide any advanced features such as removal or remediation.
- The system may not scale well for large numbers of users or files.
- The system may be vulnerable to various types of attacks such as file upload attacks, malware evasion techniques or SQL injections.

Future Works:

- Integration with other malware detection engines or services to improve detection rates and reduce false positives.
- Adding advanced features such as removal or remediation to allow for more effective malware management.
- Implementing additional security measures to protect against various types of attacks and threats.
- Improving the user interface and user experience to make it more intuitive and user-friendly.
- Optimizing the system's performance to handle larger volumes of traffic and files.

REFERENCES

- https://docs.djangoproject.com/en/4.2/intro/tutorial01/
- https://docs.clamav.net
- Mark Lutz, Learning Python, 5th Edition, O'Reilly, 2013
- Nigel George, Mastering Django, Packt Publishing, 2016.
- Leif Azzopardi and David Maxwell, Tango with Django 2, Apress, 2019