A Real Time Research Project/ Societal Related Project Report

On

Secure Email Server Deployment with Malware Protection

Submitted in fulfillment of the requirements for the award of the

Bachelor of Technology

In

Department of Computer Science and Engineering

By

J. Rashmi Chandana	**22241A0585**
N. Himaja Shivani	**22241A05A9**
N. Laya Sree	**23245A0510**
S. Poojitha	**23245A0511**

Under the Esteemed guidance of

B. Sindhuja

Assistant Professor

Department of Computer Science and Engineering

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Bachupally, Kukatpally, Hyderabad,Telangana,India,500090

2023-2024

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

CERTIFICATE

This is to certify that the Real Time Research Project/ Societal Related Project entitled **"Secure Email Server Deployment with Malware Protection"** is submitted by **J.Rashmi Chandana(22241A0585), N.Himaja Shivani(22241A05A9), N.Laya Sree(23245A0510), S.Poojitha(23245A0511)** in fulfillment of the award of a degree in BACHELOR OF TECHNOLOGY in Computer Science and Engineering during the academic year **2023-2024.**

INTERNAL GUIDE	HEAD OF THE DEPARTMENT
B. Sindhuja	**Dr. B. SANKARA BABU**
Assistant Professor	**Professor**

ACKNOWLEDGEMENT

Many people helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we wish to express our deep gratitude to our internal guide **B.Sindhuja, Assistant Professor**, Department of CSE for his/her support in the completion of our project report. We wish to express our honest and sincere thanks to **Dr. Y.Krishna Bhargavi and Ms. V. Jyothi** for coordinating in conducting the project reviews, **Dr. B. Sankara Babu, HOD,** Department of CSE for providing resources, and to the principal **Dr. J. Praveen** for providing the facilities to complete our Real Time Research Project/ Societal Related Project. We would like to thank all our faculty and friends for their help and constructive criticism during the project completion phase. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

J.Rashmi Chandana (22241A0585)
N.Himaja Shivani (22241A05A9)
N.Laya Sree (23245A0510)
S.Poojitha (23245A0511)

DECLARATION

We hereby declare that the Real Time Research Project/ Societal Related Project entitled "**Secure Email Server Deployment with Malware Protection**" is the work done during the period from **2023-2024** and is submitted in the fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering from **Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous under Jawaharlal Nehru Technology University, Hyderabad).** The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

J.Rashmi Chandana (22241A0585)
N.Himaja Shivani (22241A05A9)
N.Laya Sree (23245A0510)
S.Poojitha (23245A0511)

	Table of Contents	
Chapter	**Title**	**Page No.**

LIST OF FIGURES		
Fig. No.	**Title**	**Page No.**
4.2	System Architecture	13
4.3.1	Use Case Diagram	15
4.3.2	Class Diagram	16
4.3.3	Component Diagram	17
4.3.4	Sequence Diagram	18
5.3.1	Creating a Server	22
5.3.2	Choosing Ubuntu	22
5.3.3	Server Specifications	23
5.3.4	SSH Password	23
5.3.5	Server Connection type	24
5.3.6	Domain	24
5.3.7	SSH Connection	25
5.3.8	Set FQDN	25
5.3.9	iRedMail Installation	26
5.3.10	Run bash Command	26
5.3.11	iRedMail Server	27
5.3.12	Default directory is chosen	27
5.3.13	Password for database	27
5.3.14	Domain name	28
5.3.15	Tools	28
5.3.16	Selected Tools and packages	29
5.3.17	Mail credentials	29
5.3.18	Reboot and re-login	30
5.3.19	Certbot Installation	30
5.3.20	SSL Certificate	31
5.3.21	SMTP and IMAP Settings	31
5.3.22	Postfix Configuration	32

LIST OF TABLES

ABSTRACT

The project, "Secure Email Server Deployment with Malware Protection," tackles the critical concern of email communication security. By implementing an advanced email server with robust encryption and proactive malware detection mechanisms, such as real-time scanning and heuristic analysis, the project aims to bolster protection against threats lurking within email attachments and links. Through a comprehensive analysis of existing email server architectures and security protocols, the project tailors a deployment strategy that seamlessly integrates these security measures. By leveraging a combination of open-source and commercial security tools, the email server's malware protection capabilities are fortified.

The projected outcome is an email server infrastructure that significantly curtails malware risks, thereby promoting a more secure communication environment. We not only build a well-protected against malware threats but also promote the key of secure and honest electronic communication. This comprise implementing the latest security measures to safeguard against malicious software while ensuring that all electronic communications remain confidential and reliable. By doing this project, we promote a digital space where users can confidently share information without fear of breaches or data theft. Our project crafts a resilient e-mail server using various tools, prioritizing confidentiality and reliability amidst rising digital risks. Through this initiative, the project contributes to the broader goal of enhancing digital security amidst evolving cyber threats.

1. INTRODUCTION

In today's digital world, where communication frequently happens through mails, protecting these messages is crucial. That's where the project "Secure Email Server Deployment with Malware Protection" comes in. This project is all about making an email server that is super safe and protects your messages from unauthorized access. This email server would be built using open-source tools to make sure your emails are completely safe with the highest level of security.

The project combines different open-source tools to make the email server good at stopping harmful things from getting in. This new system uses the security to stop phishing attacks and their tracks. This project will ensure that your personal information stays private and makes sure that it does not fall into the wrong hands. This way, when we send or receive emails, we can be more confident that our information and personal data are safe from harm. Email security is the practice of providing extra protective layer of security for email accounts and communications from the software developed by hackers “cybercriminals” to steal data and destroy computer systems.

The project we are working to deploy an email server that ensures the access control to limit permissions so that email communication occurs in authentic way. A secure email server defend against phishing and malware threats and helps to prevent the data being carried away by encrypting emails containing sensitive information. The secure email server has built-in-anti-malware and anti-spam filters block malicious attachments and spam messages. The system analyzes email content ,attachments and links to detect and block suspicious or harmful messages before they reach the recipient. we use combination of different tools and software that provide very much security for the email users to avoid various problems like stealing confidential information, financial transactions and also Enhancing Security like protecting users from phishing, malware, and spam threats. The email server utilizes secure transmission protocols to encrypt data in transit, fix up protection and monitoring against man-in-the-middle attacks. The system analyzes email content and attachments to detect and block suspicious or harmful messages before they reach the recipient.

We use multiple filters to reduce the volume of spam reaching users' inboxes, improving the overall email experience. This system offers scalability to accommodate the growing needs of businesses organizations. Multiple layers of malware and spam protection are furnished to stop hazards before

they penetrate the network and establishing a decent email infrastructure that users can trust for their communication needs. This dedicated approach ensures that something unpleasant or violent will happen situations are neutralized at the earliest stage.

This project focuses on safeguarding sensitive information through encryption and secure protocols which contribute a massive enhancing security, ensuring privacy and promoting trust. The email server stay updated on the phishing methods and malicious domains, enabling real-time protection. In this system setup suspicious attachments are quarantined before they reach the user's inbox providing an additional layer of control over incoming emails.

By implementing stringent access control mechanisms, the email server restricts permissions that only authorized users can send and receive emails ensuring the privacy of email communications is paramount. We achieve this through robust encryption and secure protocols. Our massively scalable mail transfer agent offers multiple layers of malware and spam protection by stopping threats before reaching your network. By doing this, the project wants to make the internet a safer place for everyone.

2. SYSTEM REQUIREMENTS

2.1 Software Requirements

a.Operating system:

Linux distribution such as Ubuntu 22.04.4 LTS this makes an excellent option for developers and anyone looking forward by providing support for the latest hardware and improved system performance for a stable and secure operating system for long-term use.

b.Web Client:

Roundcube is a user-friendly webmail interface and its open-source nature and extensibility make it a popular choice for individuals looking for a customizable, secure, and self-hosted email solution. It provides a modern web-based interface for accessing email, similar to popular webmail services like Gmail or Outlook.com, but it can be hosted on your own server, giving you complete control over your data and email environment. It enables convenient access to e-mails while upholding security standards.

c.Web Server:

Nginx is one of the most popular web servers in the world due to its high performance, stability, rich feature set, and low resource consumption. Unlike traditional servers like Apache, Nginx uses an event-driven, asynchronous architecture, which allows it to handle many connections with a small memory footprint.

d.Security:

Fail2ban is a security tool used to protect servers from attacks and unauthorized access attempts by blocking malicious IP addresses. It works by monitoring log files for specific patterns that has suspicious malware behavior such as repeated failed login attempts, and then takes action to block the offending IP addresses for a configurable period of time for reducing the risk of attacks on the server.

e.Admin Panel:

iRedMail is an open-source mail server solution that allows users to deploy a full-featured email server on their own hardware or virtual machine. It Uses Postfix as the core mail transfer agent (MTA) for handling the sending and receiving of emails. iRedMail supports various Linux distributions, including Ubuntu and also supports for multiple domains makes it suitable for hosting providers who need to manage email services for multiple clients.

f.Database:

MariaDB is a relational database management system underpins the user authentication and system configuration required for database storage (used by tools like Postfix and other services).

g.Monitoring:

Netdata: Real-time monitoring ensures optimal performance and timely threat detection, minimizing potential vulnerabilities and it offers real-time performance monitoring and health checks for your server.

2.2 Hardware Requirements

a. CPU: 1 Core, depending on the expected email volume, a multi-core processor (e.g., Intel Xeon or AMD Ryzen) with sufficient clock speed is recommended.

b. RAM: At least 3GB RAM.

c. Memory Size: 3072 MB

d. Storage: 20 GB

e. SSD Disk: 20 GB, SSDs (Solid State Drives) for faster access times, especially for the database (MariaDB) and mail storage.

3. LITERATURE SURVEY

Title: Devising and Detecting Phishing Emails Using Large Language Models.

Authors, Year: Arun Vishwanth, Fredrik Heiding, Peter S. Park, Jeremy Bernstein, Bruce Schneier,2024.

Methodology: Models such as GPT [29] and Claude 1have demonstrated.

Observings: Our findings show that a one-size-fits-all approach is ineffective for creating phishing emails and helping users avoid being phished. Large language models are highly adept at achieving this personalization, which can be used maliciously, AI enhancement drastically reduces the cost of spear phishing attacks, sometimes rendering them as cheap as arbitrary mass-scale emails. Because of this, large language models significantly increase the incentives for launching spear phishing attacks

Shortcomes: The Limitations of this paper is reliance on AI for both attacking and defending creates an ongoing arms race, where defensive measures must continually adapt to new AI-generated threats. Finally, the economic feasibility of AI-enhanced phishing attacks raises ethical and regulatory concerns, necessitating robust frameworks to manage the misuse of such technologies.

Title: Improvement of Legitimate Mail Server Detection Method using Sender Authentication.

Authors, Year: Shuji Sakuraba, Minami Yoda ,Yuichi Sei,2021.

Methodology: This paper proposes a method by building an allowlist of legitimate senders.

Observings: It identifies forwarded emails (SPF fail, DKI pass) and extracts the sender(assumed legitimate).Then results of forwarded sender gets additional legitimate domains(excluding known spam sources). The paper utilizes a dataset of email received by an ISP mail service.

Shortcomes: Limited Dataset, Space Complexity not Addressed and the paper focuses on accuracy but doesn't consider the storage requirements.

Title: The effect of social media user behaviors on security and privacy threats.

Authors, Year: Guter Kalem, Pinar Sarisaray Boluk,2020.

Methodology: Survey method and data collection.

Observings: This paper helps us to understand how people use social media and how it effects their safety online by comparing both Turkey and Iraq. We see how culture shapes online behavior. We came to know about the differences that help us to improve online safety awareness and practices.

Shortcomes: The limitations of this page is only using surveys might not give us the full picture of how people behave online and why and also some people might not answer surveys honestly which could affect the results.

Title: On the Implementation of a Secure Email System with ID-based Encryption.

Authors, Year: Eltigani, Abdelsatir,2020.

Methodology: Identity-Based Encryption scheme, PublicKey, Generation, Extract, Encrypt ,Decrypt.

Observings: The secure email system based on IBE ensures confidentiality and authenticity of email communication, mitigating risks associated with eavesdropping and unauthorized access. The paper may discuss the practical aspects of implementing IBE in real-world email systems, including compatibility with existing email protocols and integration challenges.

Shortcomes: The implementation is not built on standard Java Cryptographic Architecture(JCA) libraries.

Title: A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach.

Authors, Year: Sanaa Kaddoura ,Omar Alfandi ,Nadia Dahmani,2020.

Methodology: Text cleaning, feature extraction(TF-IDF, Count vectorizer), model training(FFNN,BERT), evaluation(F1-score).

Observings: The paper might discuss the ability of deep learning models to generalize well to new and unseen spam email patterns, improving the adaptability of spam detection systems. Enron emails (spam/not spam) data was used.

Shortcomes: Limited feature exploration, Lack of generalizability testing such as data scarcity, model interpretability, or computational resource requirements.

Title: Improving Efficiency of E-mail Classification Through On-Demand Spam Filtering.

Authors, Year: Shafiya ,Tariq Banday,2020

Methodology: Controlled Experiment, Real-world Deployment.

Observings: In this paper we found to Deploy the on-demand filtering system to a limited set of email accounts, tracking metrics like: Spam detection rate.

Shortcomes: The Limitations of this paper is reliance on AI for both attacking and defending creates an ongoing arms race, where defensive measures must continually adapt to new AI-generated threats. Finally, the economic feasibility of AI-enhanced phishing attacks raises ethical and regulatory concerns, necessitating robust frameworks to manage the misuse of such technologies.

Title: Unveiling the connection between Malware and Pirated software in South East Asian countries.

Authors, Year: Ramkumar Rajendran, Mohammod Naveed Aman, Biplap Sikdar Asif,2024.

Methodology: Sample collection,sample imaging ,Data collection ,AV Engine.

Observings: The data collected was analyzed using statistical tools in Microsoft excel as well as open source software R. The results show that adware and Trojans are the most prevalent types of malwares in pirated software. Only limited data has been taken whereas it can do a lot better based on the domain.

Shortcomes: The focus in this paper is towards South East Asian countries but not on full population .This paper does not provide how to reduce malware associated with pirated software.

Title: Phishing or Not Phishing? A Survey on the Detection of Phishing Websites.

Authors, Year: Rasha Zieni, Maria Carla,2023.

Methodology: List based detection, Similarity based detection, Machine based detection

Observings: The paper has shown that many efforts has been dedicated to the detection of phishing websites it can be better regarding all websites, among the various detection approaches machine learning methods are becoming quite popular because of their ability to detect zero hour attack and handle efficiently newly discovered phishing web pages.

Shortcomes: Catching phishing sites well might not be possible, can be tricked by attackers and mistakes can be happened.

Title: Email Security concept, Formulation and Applications.

Authors, Year: Abdullah Hussein AI-Ghushami, Dubeeruddin syed, Ameena Zainab, Haya Abdelshahid,2022.

Methodology: Cryptographic Algorithms, Protocol Analysis, Mathematical Concepts and Security principle evaluation.

Observings: This paper focus on evaluating the security, effectives, performance and protocols for protecting email communications.

Shortcomes: Research on cryptographic security concepts behind e-PHP and S/MIME email protocols have some disadvantages such as limited scope ,lack of practical implementation and complexity.

Title: Malicious URL Detection :A Comparative Study.

Authors, Year: Shanta nu, Janet ,R Joshua Arul Kumar,2021.

Methodology: Machine learning classification is used for detection.

Observings: This paper has used public dataset from Kaggle comprising 450,000 URLs.

Shortcomes: The paper doesn't mention the metrics used to evaluate the model's performance.

Title: Malware detection using Byte Streams of different File Formats.

Authors, Year: Young-seob Jeong ,Sang-Mi Lee, Jong-Hyun Lee,2022.

Methodology: Malcon V model trained with HWP+PDF streams ,here they used CNN models for measuring the method.

Observings: This paper aims that solving the malware detection task that is basically a binary classification ,here developed a model that predicts a label malware or benign ,this employed the cost-sensitive learning technique.

Shortcomes: This paper is succeeded in malware detection using Malcon V model but failed by the model trained with only HWP and PDF streams and tried to feed only PDF byte streams to the HWP malware detection model and its performance is very low.

Title: A Boosting-Based Hybrid Feature selection and Multi-Layer Stacked Ensemble Learning Model To Detect Phishing Websites.

Authors, Year: Laxshamana Rao Kalabange ,Srinivasa Rao,2023.

Methodology: To detect and prevent phishing attacks various methods such as Black List, Feature Extraction and Machine Learning.

Observings: This study proposes a novel hybrid feature selection approach and a boosting -based-multi layered stalking ,Ensemble Learning Model to address the challenges of detecting phishing attacks accurately.

Shortcomes: Complexity, Limited scope and Lack of Real-World Application.

Title: An Encrypted Cloud Email Searching and Filtering Scheme Based on Hidden Policy Ciphertext-Policy Attribute-Based Encryption With Keyword Search.

Authors, Year: Jian Gao , Fucai Zhou,2021.

Methodology: Dual System Encryption methodology**,** Scheme Model, Scheme Description, Security Defination and in this paper conducted the experiments using real-world email dataset. The dataset used for the experiments is BC3-Email Corpus.

Observings: This paper introduced a new solution for searching and filtering encrypted cloud email based on HPCPABKS. More extensions can be made to the scheme to realize the function of virus email protection in the future. It can be easily extended to other application scenarios, such as the searching and filtering encrypted file systems.

Shortcomes: The scheme achieves full security proved by using dual system encryption methodology and can resist offline KGA,because of the use of composite order bilinear groups, the performance of our scheme is limited. In the future, we need to improve the scheme further to make it more rapid and straightforward without reducing the security. In addition, our next work will also focus on multi keyword search and other query expression capabilities.

Title: A Secure Methodology for Filtering Spam & Malware in E-Mail System and Secure E-mail Testbed Setup.

Authors, Year: Sanjay Adiwal, Akanksha Gupta, Balaji Rajendra, B S Bindhunmandhava, 2021.

Methodology: Testbed setup is done by creating an environment to simulate real-world e-mail traffic.

Observings: The achievement of 95% accuracy in spam filtering and malware compared to normal email system configuration. This indicates the proposed system significantly improves the security of e-mail system.

Shortcomes: The results obtained in a testbed environment may not fully translate to real-world scenarios with more diverse e-mail traffic and attack vectors.

Title: **A Solution for Secure Multi-Party Certified Electronic Mail Using Blockchain.**

Authors, Year: M. Francisca Hinarejos, Josep-Lluis Ferrer-Gomila,2020.

Methodology: Used the blockchain to achieve the fairness requirement without TTP for multi-party certified email while achieving the requirement of confidentiality.

Observings: In our proposal, we have defined a solution to be used in a multi-party scenario, but this can also be used for single exchanges with one recipient, the paper only requires the execution of blockchain functions in the case of conflict, and demonstrated that if the execution of the functions is necessary, the cost is very reasonable. Finally, highlighted that they have designed their proposal considering easy integration into the standardized infrastructure of Internet email.

Shortcomes: The presented paper improves the deficiencies detected in the only previous reference we have found for multi-party certified email based on the blockchain. This improvement translates into true compliance with the confidentiality requirement, greater ease for users and a reduction in costs as they will only have to face the possible costs of executing the functions in the blockchain but not for the services of a TTP.

4.PROPOSED MODEL, MODULES DESCRIPTION UML DIAGRAMS

4.1 Proposed Model

The project "Secure Email Server Deployment with Malware Protection" focuses on enhancing email security through advanced measures.

1.Cost-Effectiveness: Utilizes open-source software for cost-effective security enhancement.
2.Resource Efficiency: Operates efficiently, minimizing resource consumption while enhancing security.
3.Reduced Spam and Phishing: Implements SPF to reduce spam and phishing threats.
4.Interoperability: Works alongside SPF and DMARC for comprehensive email security.
5.Reliability: Utilizes MariaDB for reliable storage of sensitive email data.
6.Performance: Offers high-performance database capabilities for efficient email traffic handling.
7.Ease of Use: Netdata features a user-friendly interface and intuitive dashboard, making it accessible to administrators with varying levels of technical expertise, thereby facilitating effective email security monitoring and management.
8.Email Authentication: DKIM adds a digital signature to email messages, allowing the receiving email server to verify the message's authenticity and integrity. This helps authenticate the sender's identity and ensures that the email has not been tampered with during transmission.
9.Scalability: Fail2ban is scalable and can be deployed across distributed email infrastructure, ensuring consistent security enforcement across multiple email servers and services.
10.Integration: It can be easily integrated with other monitoring and security tools, allowing administrators to correlate email security events with system performance metrics for more comprehensive analysis and response.

The project's uniqueness lies in combining software and tools to provide an additional layer of email security.

4.2 System Architecture

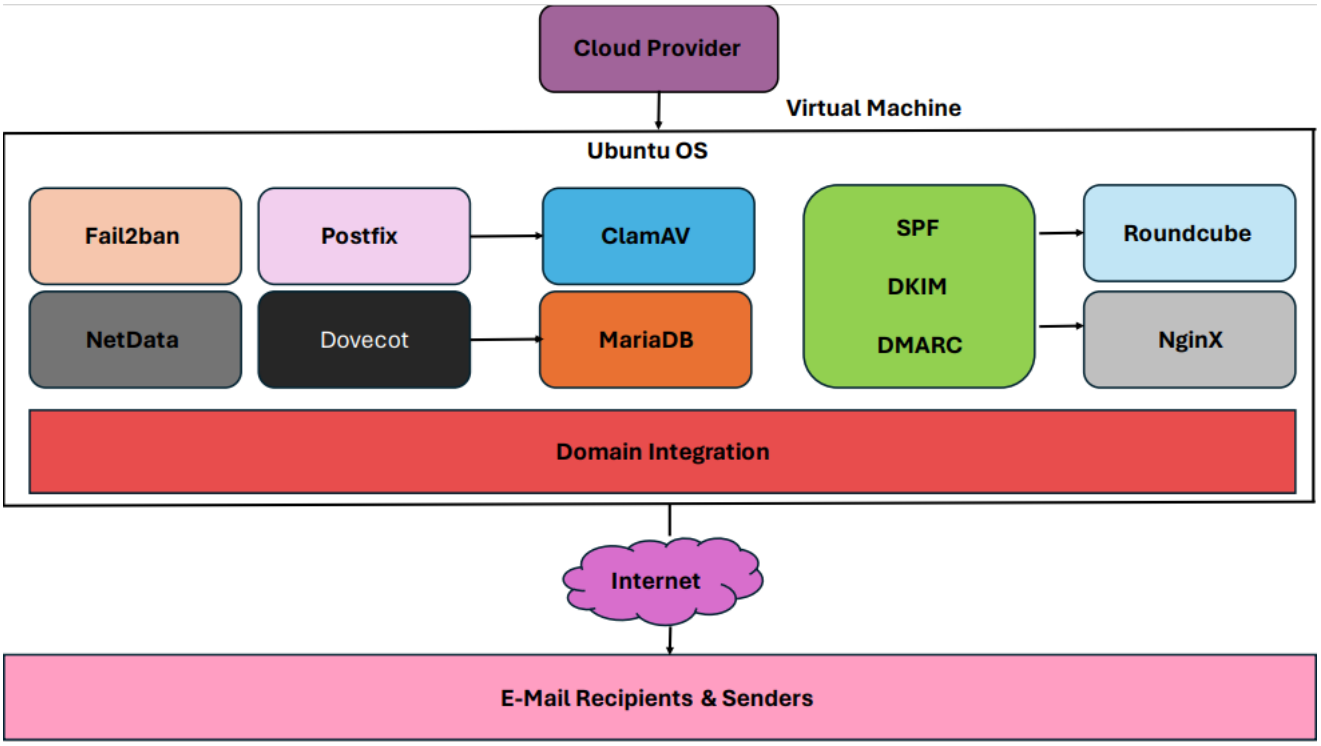

Fig.4.2 System Architecture

The System Architecture of email server uses multiple layers to make safer email communication. In the fig.4.2, the cloud provider hosts the VM, which connects to the internet to make it easy for sending and receiving emails. It also explains how various mail server components which are connected to each other for much higher security for email messages.

4.2.1 Modules Description

1.VM: Virtual Machine is like a physical computer, it includes CPU, Memory, storage which can help OS run easily on it.
2.Ubuntu: Ubuntu is a popular Linux which can install and manage software applications easily.
3.Fail2ban: Fail2ban is a security tool that runs in the background of an email server system. it is used for protecting our emails from any unauthorized access by blocking suspicious Ip addresses.
4.ClamAV: ClamAV comes under mail server security that is used for checking viruses in out emails. it integrates with postfix for security filtering.
5.MariaDB: MariaDB is a database used to store information about the user and it also configures necessary settings for the email server.
6.SPF, DKIM, DMARC: These are three authentication mechanisms which validate email authenticity and prevent phishing attacks.
7.Postfix: postfix is an open-source mail transfer agent (SMTP server) which is responsible for sending and receiving mails.
8.Dovecot: Dovecot is an open-source mail delivery agent (IMAP and POP3 server) which acts as an email inbox where all emails are stored until you read them.
9.Nginx: Nginx is an open-source web server software that helps to manage the admin panel.
10.Netdata: Netdata is used for monitoring system health.
11.Roundcube: Roundcube is a webmail interface that helps us to access our emails from any computer.

The above tools together create a secure email server system. They provide a solution for managing and securing an efficient email server system. They address various aspects of email management, like security, storage, and user access making sure the email system works smoothly and safely.

4.3 UML Diagrams

4.3.1 Use Case Diagram

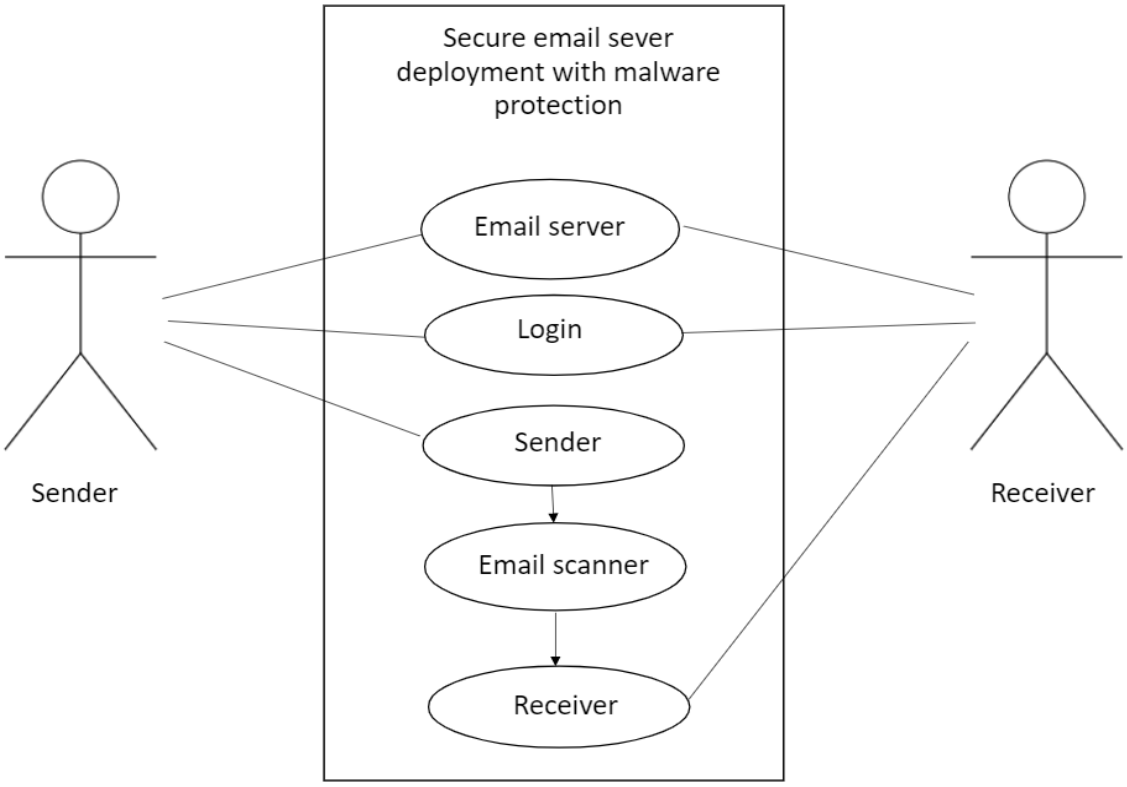

Fig.4.3.1 Use Case Diagram

The Use Case diagram represents the process of sending and receiving the email in a secure email server setup that includes malware protection.
The Email server and Email scanner are the System Components used in this Use Case diagram.

The sender enters their username and password to access their email account, then the Email server uses an email scanner to check the email for viruses or harmful content. If no harmful content is found, the email is delivered safely to the receiver's inbox.

4.3.2 Class Diagram

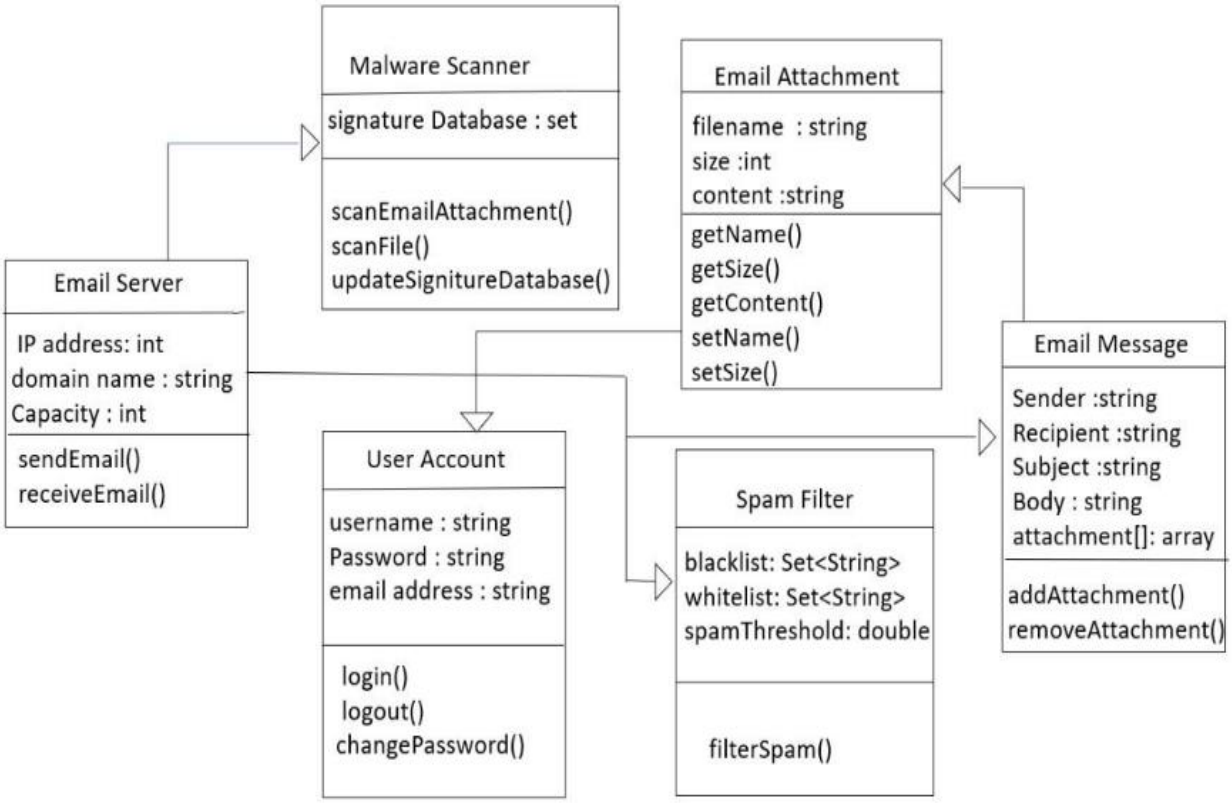

Fig.4.3.2 Class Diagram

The class diagram illustrates the various key components and their interactions in an email system. Each class contains data fields and methods relevant to the email system.

1.Email server contains IP address of the server, domain name, storage capacity of the server.
2.Email Attachment contains filename, size of the file, content in the file.
3.Spam Filter contains blacklist email addresses, set of whitelist email addresses, threshold for marking emails as spam.
4. Malware Scanner contains Database of known malware signatures (set).
5. Email Messages contain sender, receiver, subject, body , attachment .
6.User Account contains username, password, email address.

This class diagram shows that each component works together to provide a secure email system.

4.3.3 Component Diagram

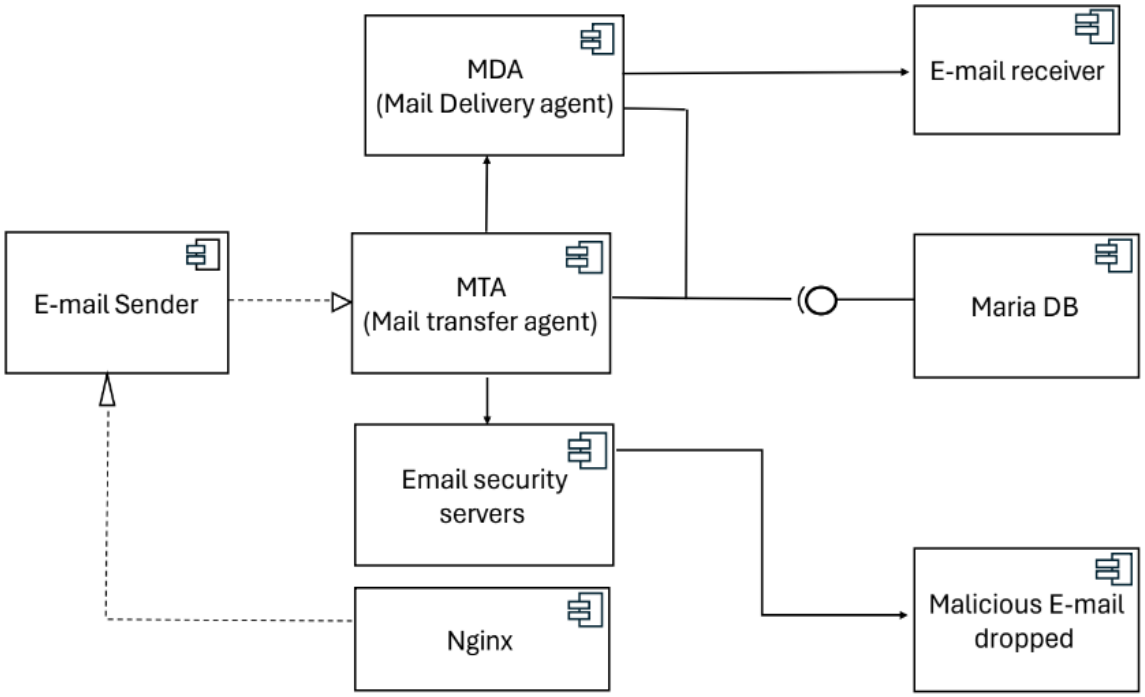

Fig.4.3.3 Component Diagram

The component diagram illustrates the different components which interact with each other in order to provide security for an email system.

Components –

1.MTA (Mail Transfer Agent): MTA handles the transfer of emails from sender to the receiver. Postfix is used as MTA in this email system.

2.MDA (Mail Delivery Agent): MDA is used to deliver email safely to the receiver's email box. Dovecot is used as MDA in this email system.

3. MariaDB: MariaDB is Database system used to store email configuration data.

4. Email security servers: These servers scan the malicious content in the emails. ClamAV is used for malware scanning which comes under email security server.

5.Nginx: Nginx serves the web-based interface for users to access their emails, such as Roundcube.

6.Emails which are identified as malicious by the email server system are dropped and not delivered.

4.3.4 Sequence Diagram

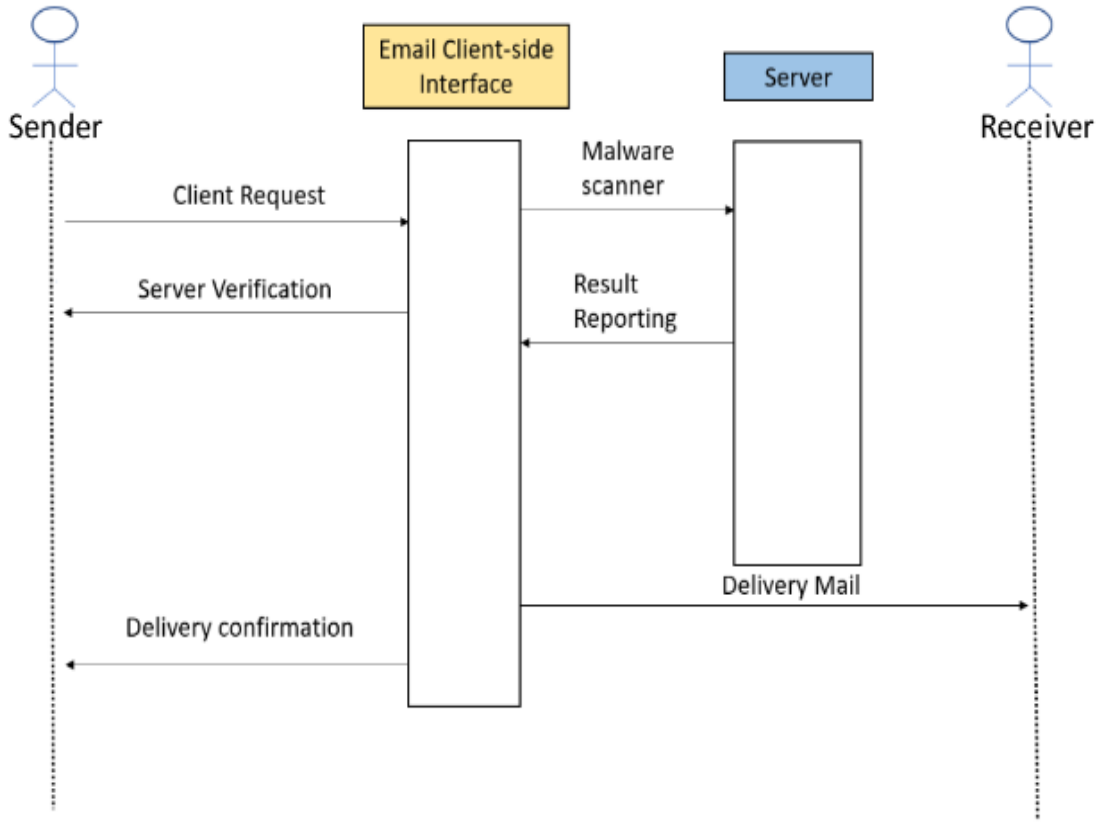

Fig.4.3.4 Sequence Diagram

The Sequence diagram illustrates how the email is sent and received with high security.
When an email is sent to the receiver, before it delivers , it makes sure to pass security checks by the email server system .
When all the security checks are passed by malware scanner, the server double-checks the email to make sure everything is in order to send it to recipient . After the server's verification, email is safely delivered to the recipient.

5.IMPLEMENTATION, EXPERIMENTAL RESULTS &TEST CASES

5.1 Preparations

a) Server Setup:

1.Create a server on a cloud platform and choose an OS image (ubuntu 22.04 Version 64-bit LTS)

2.Set username as root user and password to access the root user accordingly.

3.Connection Type of this server should be SSH Connection.

4.Make sure you have public Ip address for server.

5.Make sure to keep the state of the server running.

b) Domain Name:

1)Create a domain name on any website. The domain used in this project is “rtrpcyber.xyz” .

2)Make sure that the domain status is active after it’s created .

c)Hostname:

1) Set a fully qualified domain name (FQDN) hostname on server .

2) Hostname of domain rtrpcyber.xyz is “mail.rtrpcyber.xyz”.

d)DNS Records:

1)Set up an mx record with name “mail.rtrpcyber.xyz” on the server. Priority value can be between 1 – 10.

2)Set up a record on the domain website. A record should be pointed to Ip address the server.

5.2 Linux Commands

1) SSH Connection:

Once the server is created, connect to the terminal with Linux terminal using the following command: **ssh@ root(Ip-address)**.

Initially root is used to connect to the server because the server's username is root.

After the command, you'll be asked to enter the password. Enter the password you created while creating a server.

2)Update apt repositories and add username:

a) Run command: "**sudo apt-get update**" to update apt repository information.

b) Add another username for your server to access it easily. sudo is required to manage the server rather than using the default root user every time.

To add another username, use the command "**adduser username**"

3)Set hostname:

Set "mail.rtrpcyber.xyz" as the hostname on your server. To set the hostname, add Ip address and FQDN in hostname configuration file.

Open the configuration of hostname with command: "**sudo nano /tec/hosts**". Once you successfully set the hostname , check by command "**hostname -f'** . If the hostname is displayed in the terminal, that means you successfully set your hostname.

4) iRedMail Installation:

a) iRedMail will automatically install mail server components though the manual installation is required for some tools.

b) After installing, extract the iRedMail file in your ubuntu with command: **tar xvf 1.6.8.tar.gz** .

To know the name of the tar files of yours, type **ls** command in your terminal.

c) The shell script will be in the directory of iRedMail file.

Give permission to shell script with command: **chmod +x iRedMail.sh**.

d) Finally run bash command: **sudo bash iRedMail.sh** to execute the shell script.

Follow the on-screen instructions further to complete configuring iRedMail file.

e) The credentials of our admin panel will be displayed at the last to login into our iRedAdmin panel and also roundcube credentials will be displayed along with URL's.

5) Certificate:

Obtain SSL Certificate with the help of certbot and Nginx. Nginx will act as an authenticator for the server.

6) Configuration files:

Edit the configuration files of postfix, dovecot, SSL configuration files to make sure that all are pointed to our FQDN and Ip address of our server.

Ensure that all ports are open while configuring files.

7) Webmail login:

Login to the Roundcube web mail to access the email account.

5.3 Screenshots

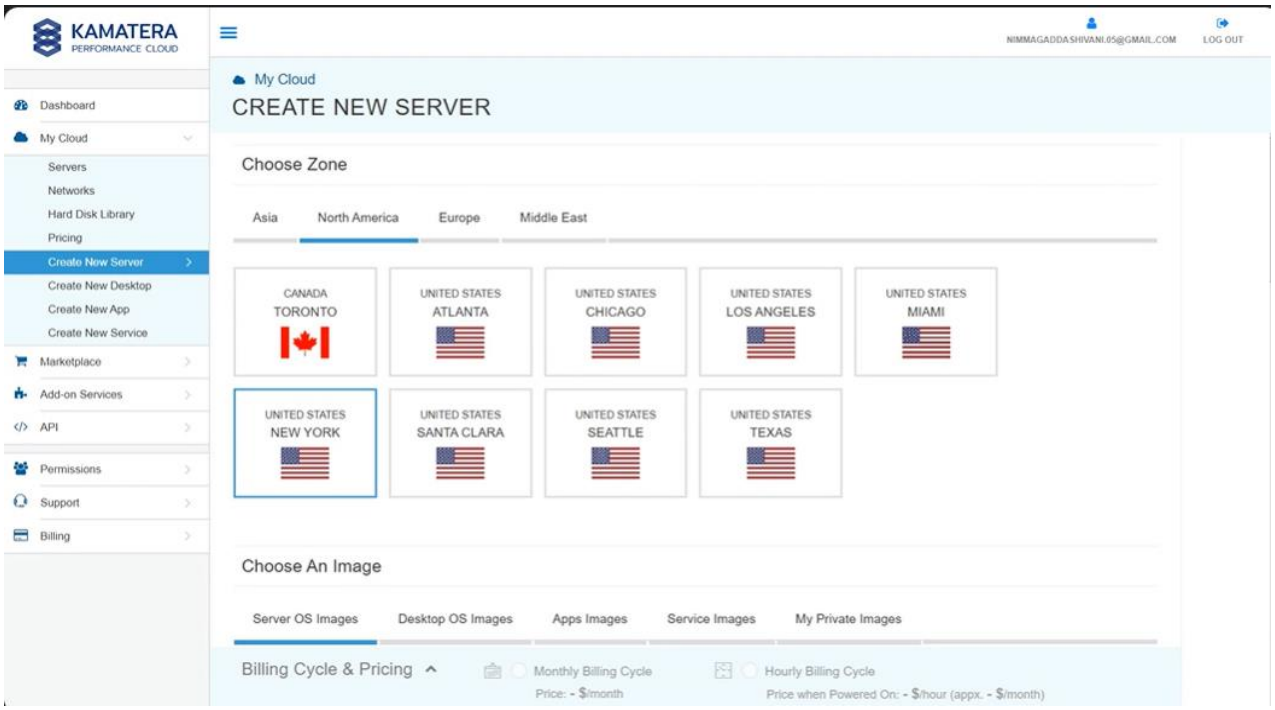

Fig.5.3.1 Creating a Server.

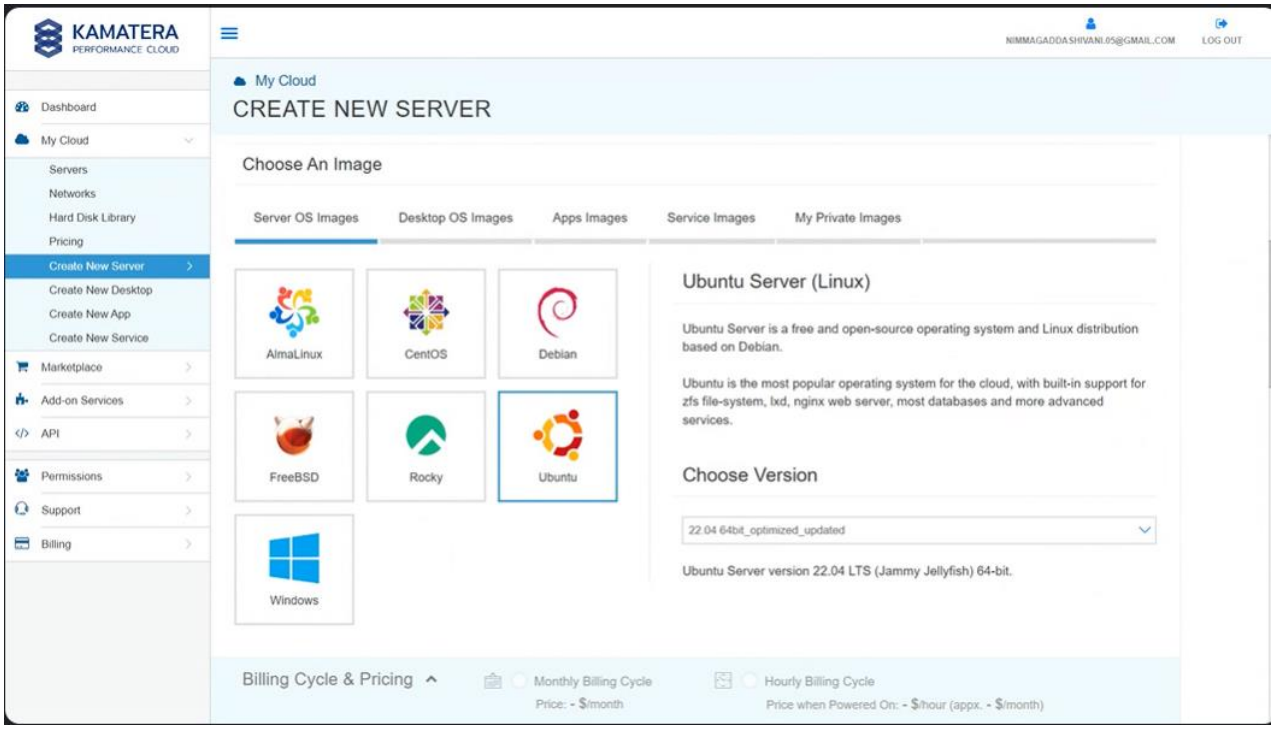

Fig.5.3.2 Choosing Ubuntu

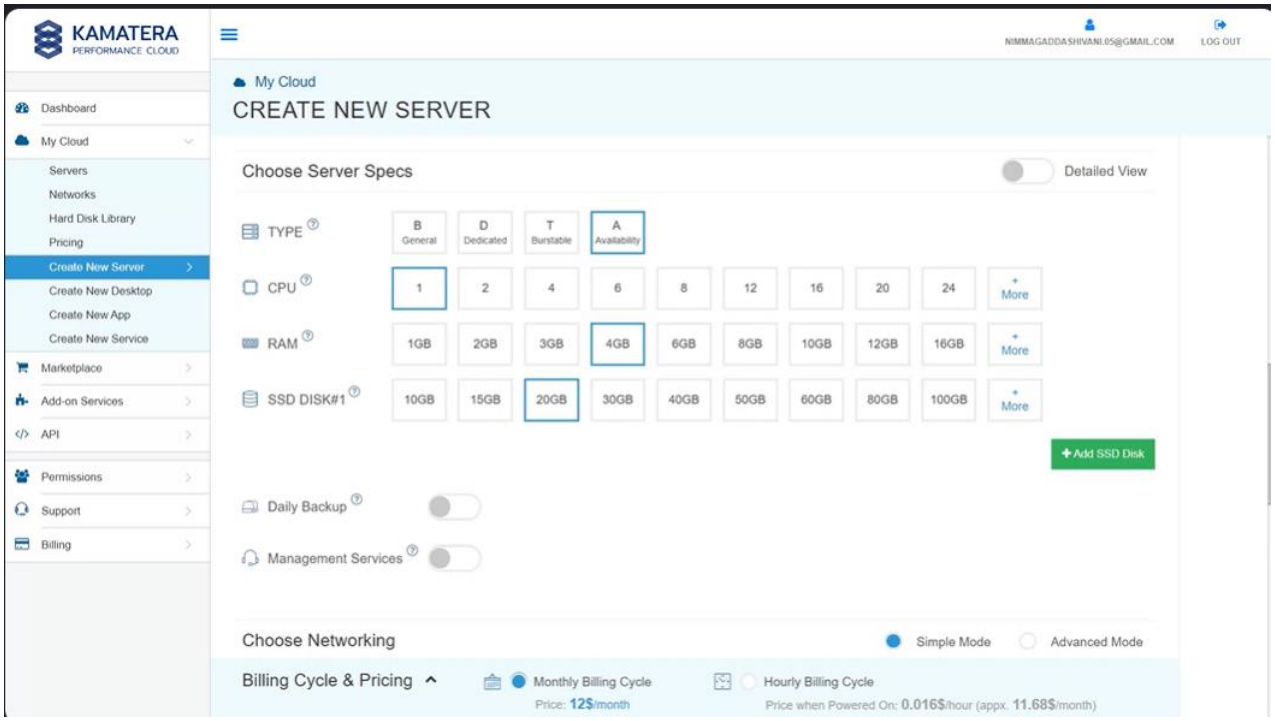

Fig.5.3.3 Server Specifications

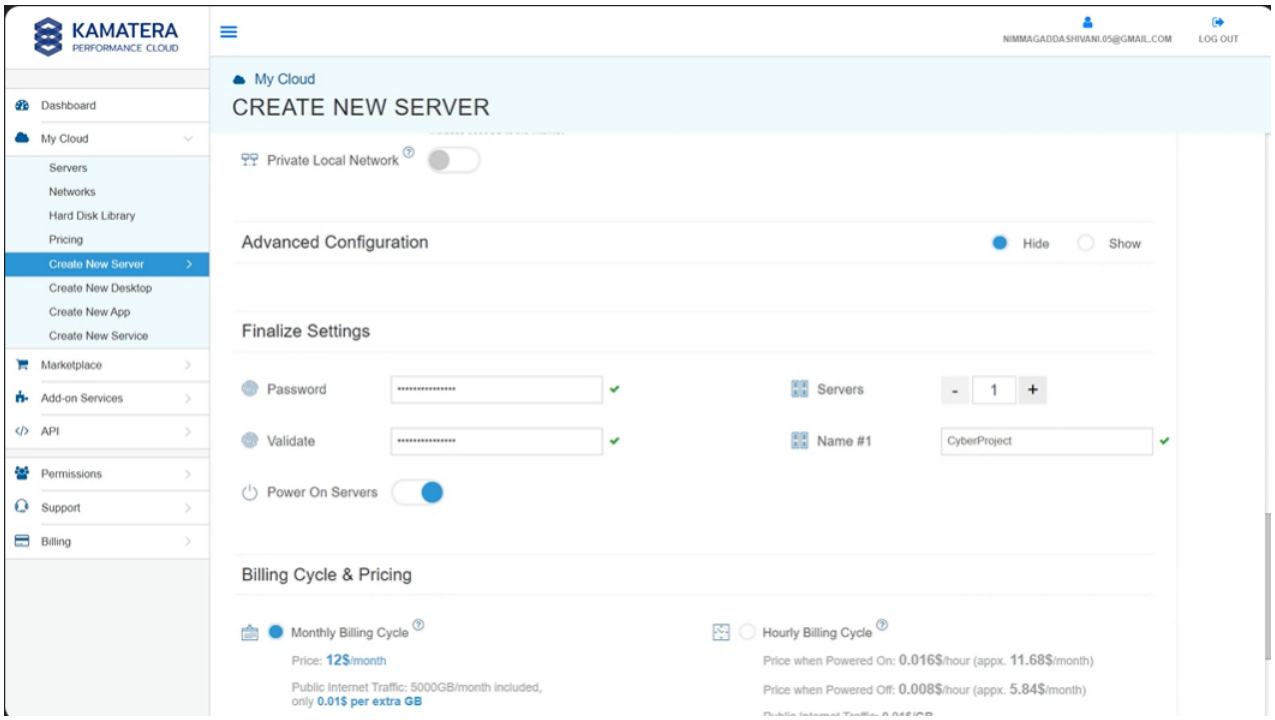

Fig.5.3.4 SSH Password

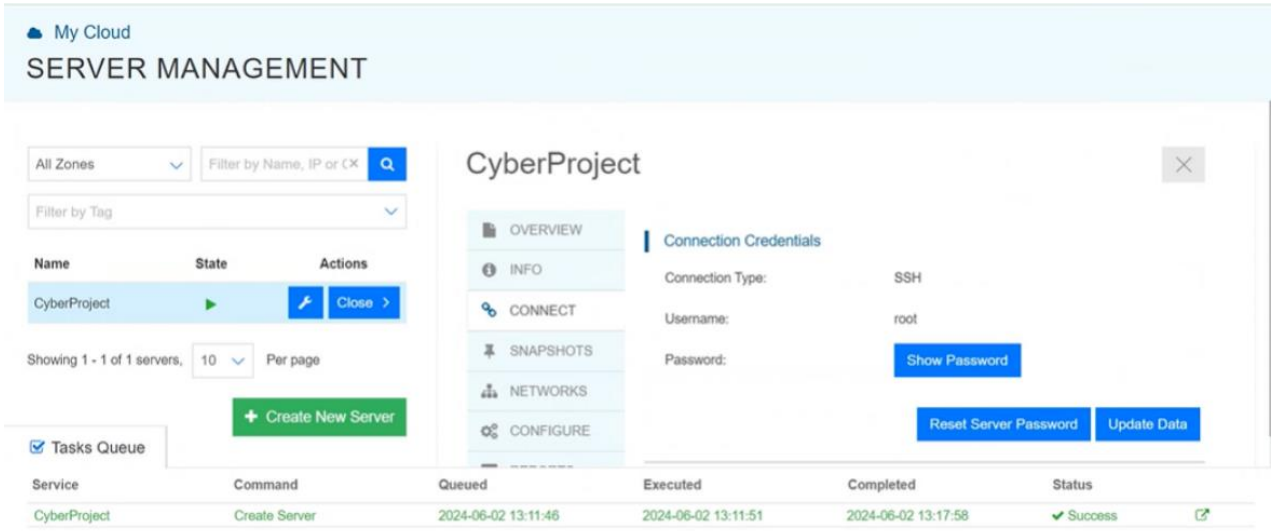

Fig.5.3.5 Server Connection Type

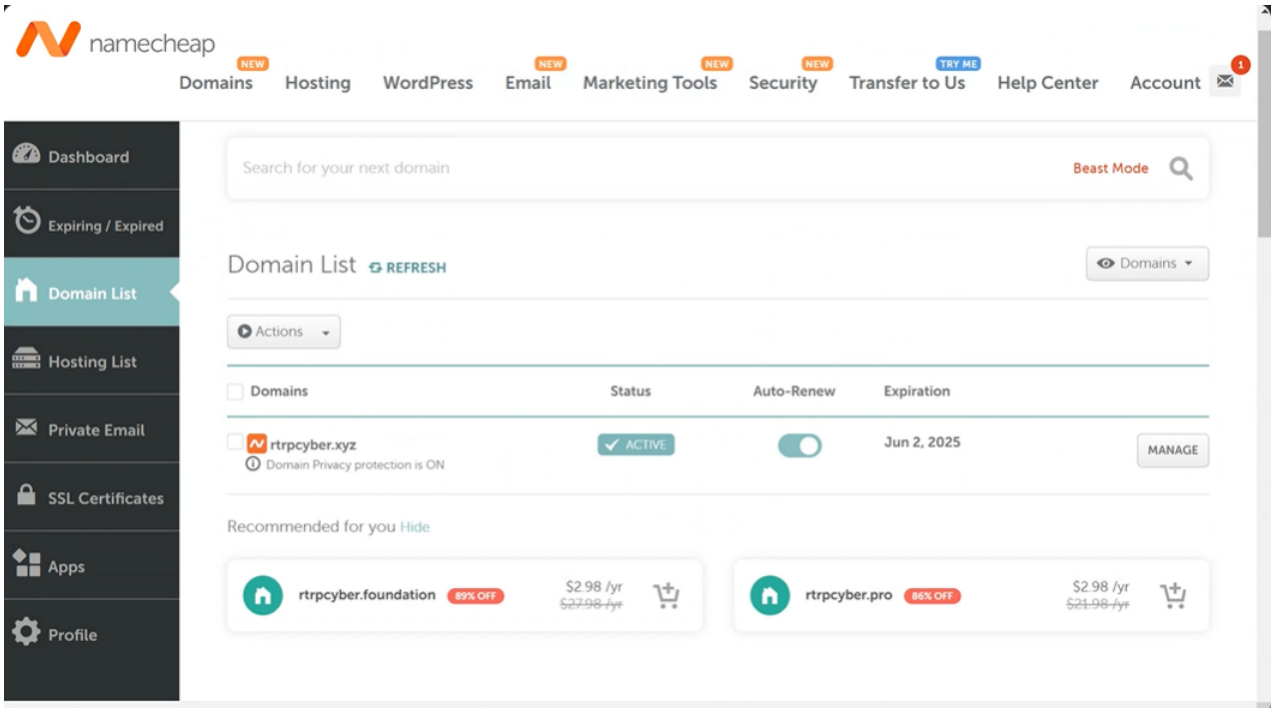

Fig.5.3.6 Domain

```
cyberpunk@cyberpunk-VirtualBox:~$
cyberpunk@cyberpunk-VirtualBox:~$ ssh root@103.54.56.183
root@103.54.56.183's password:

Permission denied, please try again.
root@103.54.56.183's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-107-generic x86_64)

 System information as of Mon Jun  3 03:40:58 PM EDT 2024

  System load:  0.0                Processes:             103
  Usage of /:   24.0% of 19.58GB   Users logged in:       0
  Memory usage: 6%                 IPv4 address for eth0: 103.54.56.183
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


root@cyberproject:~#
```

Fig.5.3.7 SSH Connection

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

cyberpunk@cyberpunk-VirtualBox:~$ ssh cyberpunk@103.54.56.183
cyberpunk@103.54.56.183's password:
Permission denied, please try again.
cyberpunk@103.54.56.183's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-107-generic x86_64)

 System information as of Mon Jun  3 04:09:38 PM EDT 2024

  System load:  0.0                Processes:             105
  Usage of /:   24.0% of 19.58GB   Users logged in:       0
  Memory usage: 6%                 IPv4 address for eth0: 103.54.56.183
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

cyberpunk@mail:~$ hostname -f
mail.rtrpcyber.xyz
cyberpunk@mail:~$ wget https://github.com/iredmail/iRedMail/archive/1.6.8.tar.gz
--2024-06-03 16:13:03--  https://github.com/iredmail/iRedMail/archive/1.6.8.tar.gz
Resolving github.com (github.com)... 140.82.112.4
Connecting to github.com (github.com)|140.82.112.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/iredmail/iRedMail/tar.gz/refs/tags/1.6.8 [following]
```

Fig.5.3.8 Set FQDN

```
cyberpunk@mail:~$ hostname -f
mail.rtrpcyber.xyz
cyberpunk@mail:~$ wget https://github.com/iredmail/iRedMail/archive/1.6.8.tar.gz
--2024-06-03 16:13:03--  https://github.com/iredmail/iRedMail/archive/1.6.8.tar.gz
Resolving github.com (github.com)... 140.82.112.4
Connecting to github.com (github.com)|140.82.112.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/iredmail/iRedMail/tar.gz/refs/tags/1.6.8 [following]
--2024-06-03 16:13:04--  https://codeload.github.com/iredmail/iRedMail/tar.gz/refs/tags/1.6.8
Resolving codeload.github.com (codeload.github.com)... 140.82.113.10
Connecting to codeload.github.com (codeload.github.com)|140.82.113.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: '1.6.8.tar.gz'

1.6.8.tar.gz                        [ <=>                                                ] 240.39K  --.-KB/s    in 0.03s

2024-06-03 16:13:04 (9.30 MB/s) - '1.6.8.tar.gz' saved [246158]
```

Fig.5.3.9 iRedMail Installation

```
iRedMail-1.6.8/update/ldap/README.md
iRedMail-1.6.8/update/ldap/update-ldap-dovecot-2.3.py
cyberpunk@mail:~$ cd iRedMail-1.6.8/
cyberpunk@mail:~/iRedMail-1.6.8$ chmod +x iRedMail.sh
cyberpunk@mail:~/iRedMail-1.6.8$ sudo bash iRedMail.sh
[sudo] password for cyberpunk:
[ INFO ] Checking new version of iRedMail ...
[ INFO ] Installing package(s): gnupg2 dialog
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  dmeventd libdevmapper-event1.02.1 libisns0 liblvm2cmd2.03 libopeniscsiusr squashfs-tools thin-provisioning-tools
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  dialog gnupg2
0 upgraded, 2 newly installed, 0 to remove and 2 not upgraded.
Need to get 309 kB of archives.
After this operation, 1311 kB of additional disk space will be used.
Get:1 http://mirror.us-ny2.kamatera.com/ubuntu jammy/universe amd64 dialog amd64 1.3-20211214-1 [303 kB]
Get:2 http://mirror.us-ny2.kamatera.com/ubuntu jammy-updates/universe amd64 gnupg2 all 2.2.27-3ubuntu2.1 [5548 B]
Fetched 309 kB in 0s (7799 kB/s)
Selecting previously unselected package dialog.
(Reading database ... 109847 files and directories currently installed.)
Preparing to unpack .../dialog_1.3-20211214-1_amd64.deb ...
Unpacking dialog (1.3-20211214-1) ...
Selecting previously unselected package gnupg2.
Preparing to unpack .../gnupg2_2.2.27-3ubuntu2.1_all.deb ...
Unpacking gnupg2 (2.2.27-3ubuntu2.1) ...
Setting up gnupg2 (2.2.27-3ubuntu2.1) ...
Setting up dialog (1.3-20211214-1) ...
Processing triggers for man-db (2.10.2-1) ...
NEEDRESTART-VER: 3.5
NEEDRESTART-KCUR: 5.15.0-107-generic
NEEDRESTART-KEXP: 5.15.0-107-generic
NEEDRESTART-KSTA: 1
Adding component(s) 'multiverse' to all repositories.
Press [ENTER] to continue or Ctrl-c to cancel.
```

Fig.5.3.10 Run Bash command

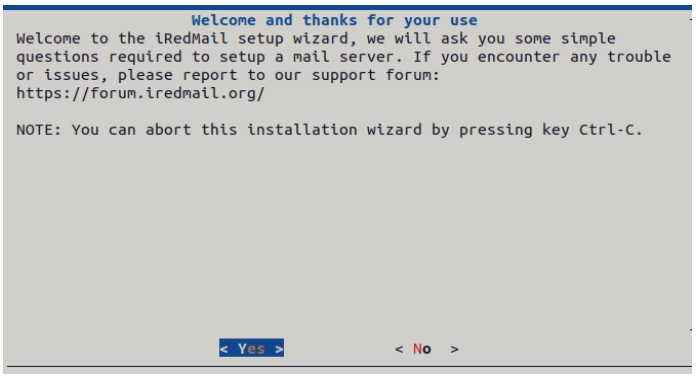

Fig.5.3.11 iRedMail Server

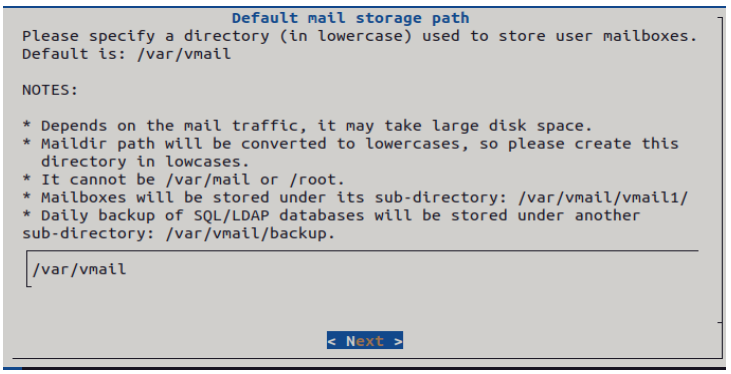

Fig.5.3.12 Default directory is chosen

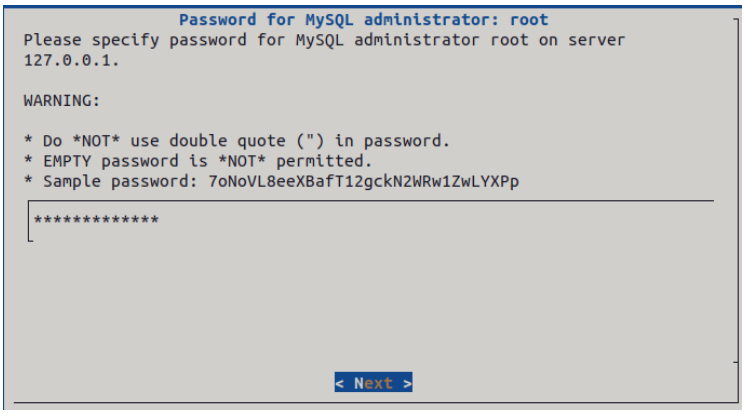

Fig.5.3.13 Password for database

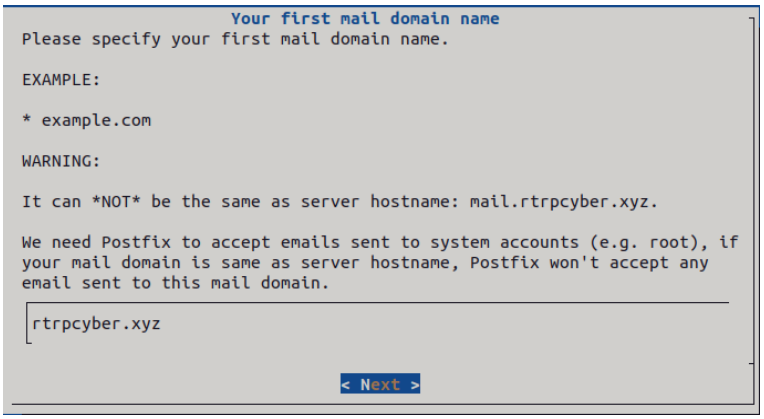

Fig.5.3.14 Domain Name

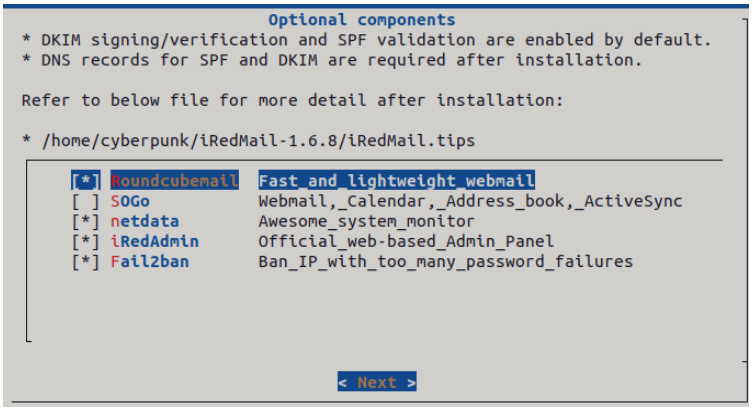

Fig.5.3.15 Tools

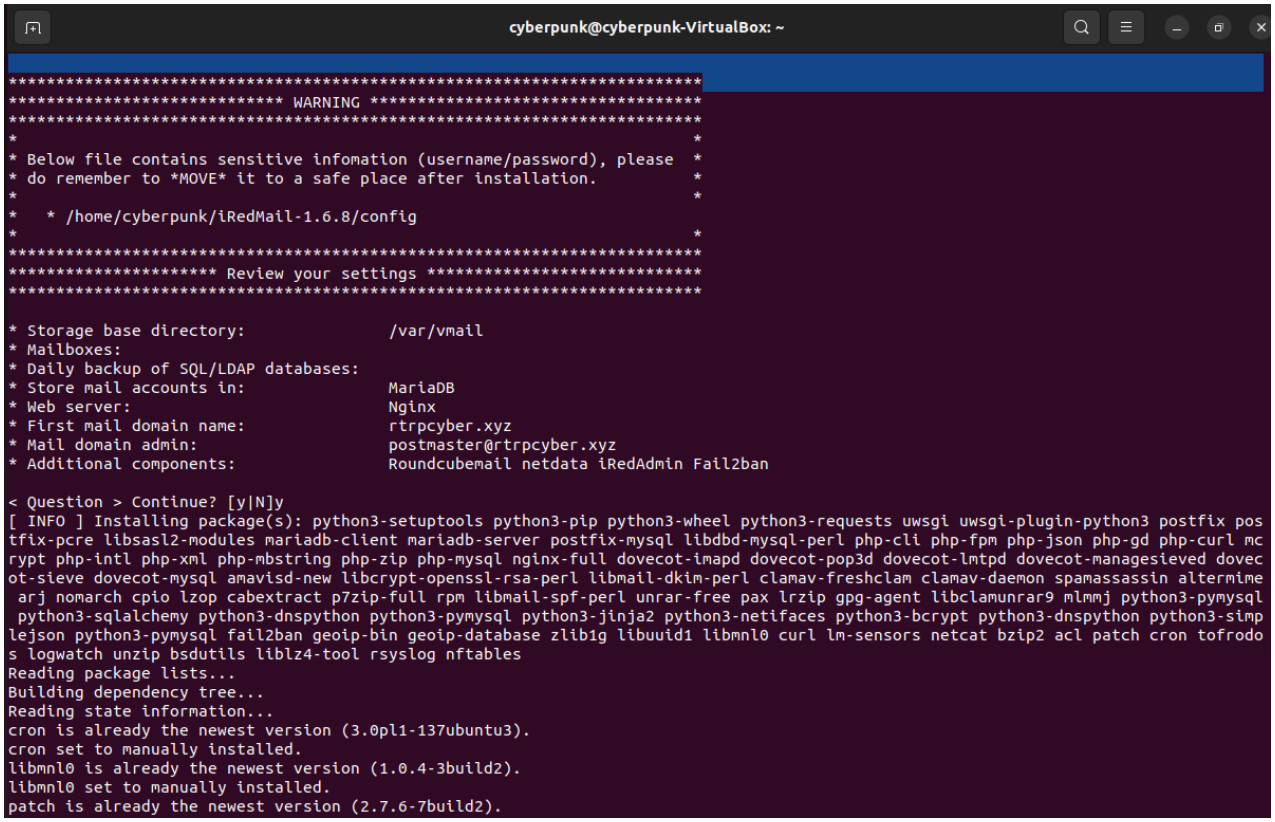

Fig.5.3.16 Selected Tools and packages

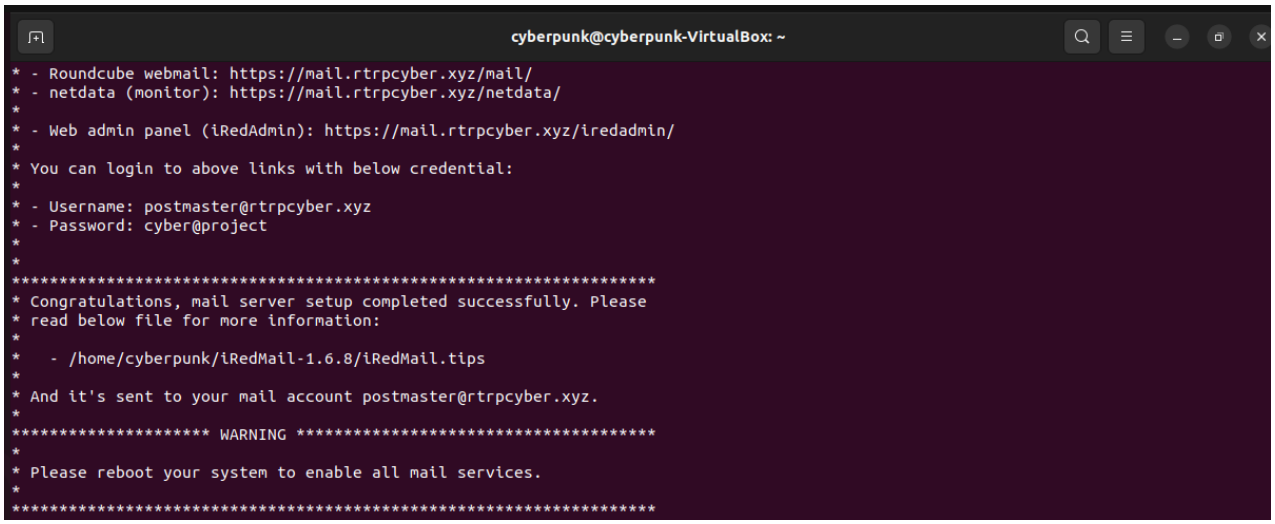

Fig.5.3.17 mail credentials.

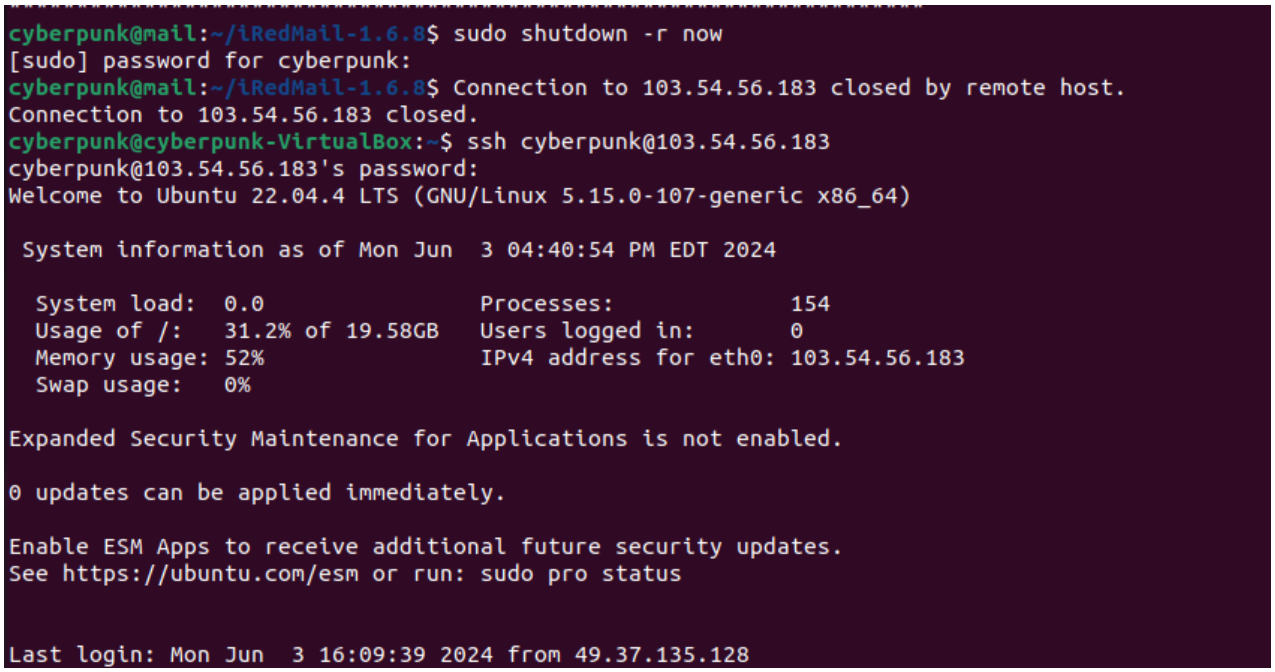

Fig.5.3.18 Reboot and re-login.

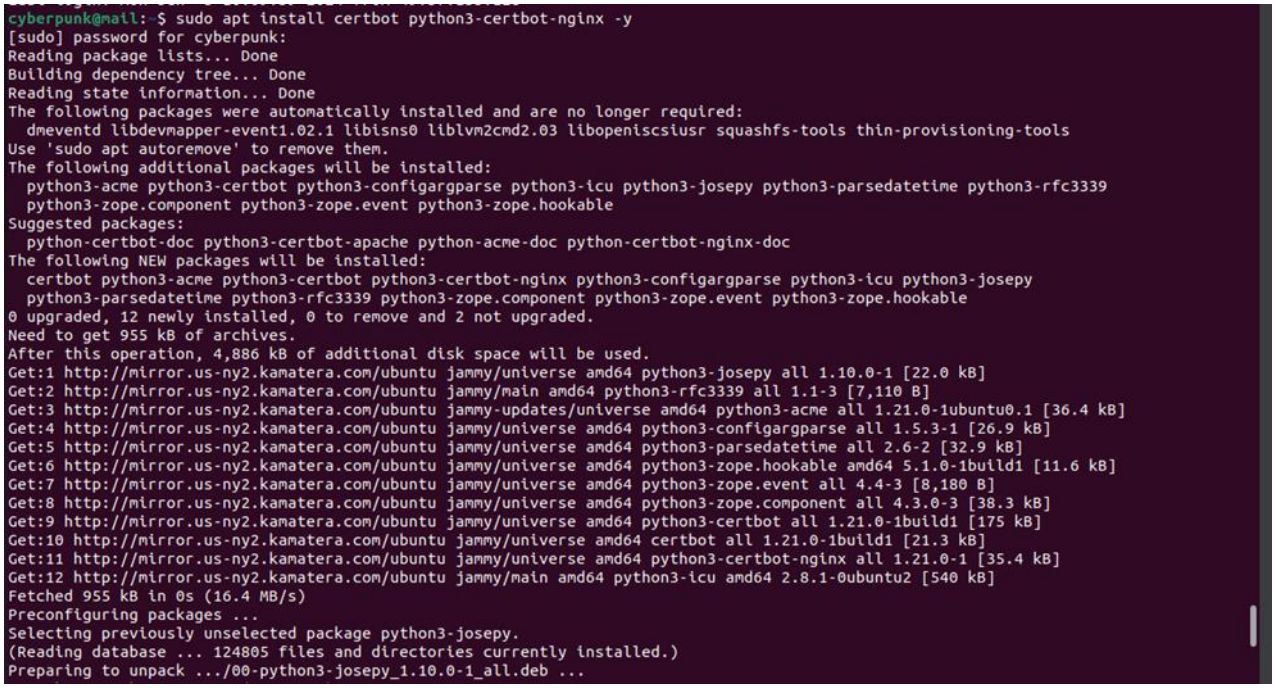

Fig.5.3.19 Certbot Installation

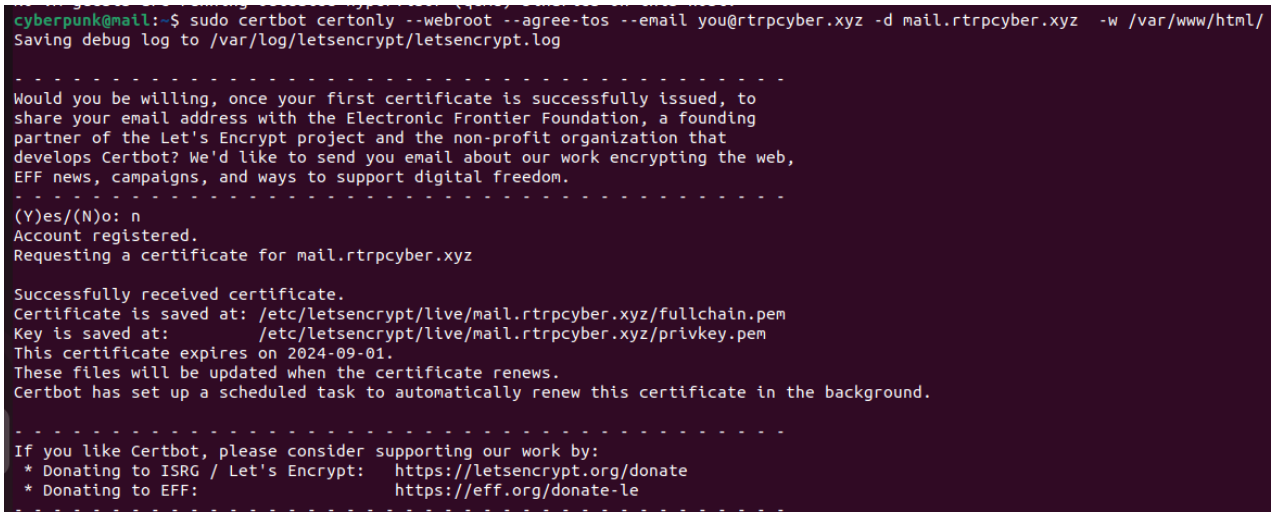

Fig.5.3.20 SSL Certificate

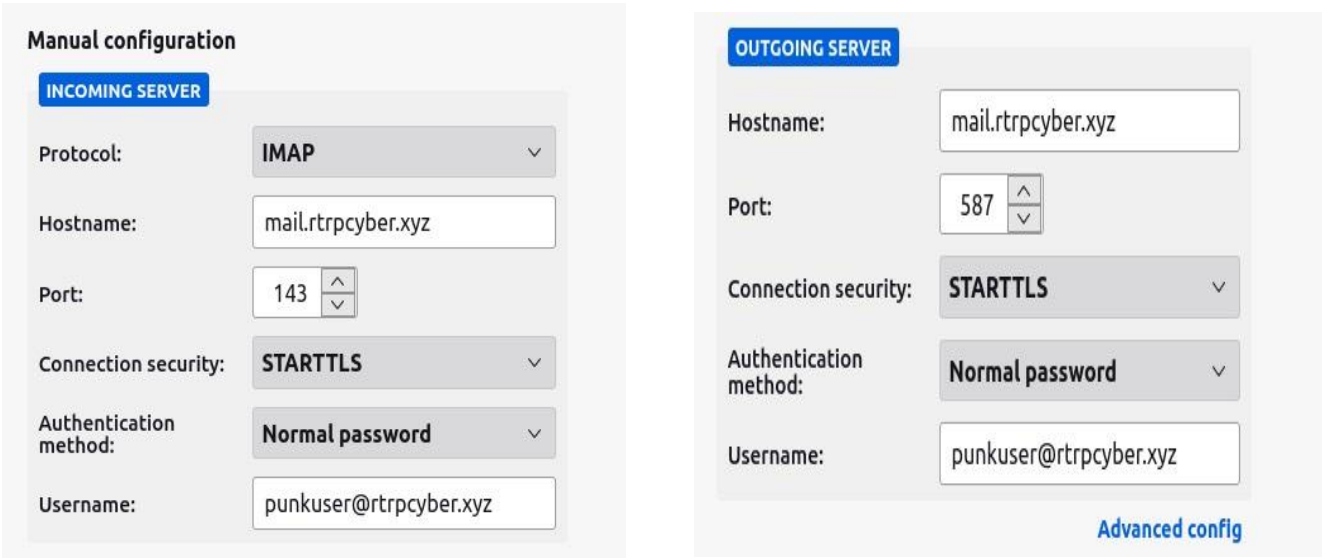

Fig.5.3.21 SMTP and IMAP Settings

```
  GNU nano 6.2                                  /etc/postfix/main.cf *
smtputf8_enable = no

#
# TLS settings.
#
# SSL key, certificate, CA
#
smtpd_tls_key_file = /etc/letsencrypt/live/mail.rtrpcyber.xyz/privkey.pem
smtpd_tls_cert_file = /etc/letsencrypt/live/mail.rtrpcyber.xyz/cert.pem
smtpd_tls_CAfile = /etc/letsencrypt/live/mail.rtrpcyber.xyz/chain.pem
smtpd_tls_CApath = /etc/ssl/certs

#
# Disable SSLv2, SSLv3, TLSv1, TLSv1.1.
#
smtpd_tls_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1
smtpd_tls_mandatory_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1

smtp_tls_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1
smtp_tls_mandatory_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1

lmtp_tls_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1
lmtp_tls_mandatory_protocols = !SSLv2 !SSLv3 !TLSv1 !TLSv1.1

#
# Fix 'The Logjam Attack'.
#
smtpd_tls_dh512_param_file = /etc/ssl/dh512_param.pem
smtpd_tls_dh1024_param_file = /etc/ssl/dh2048_param.pem

tls_random_source = dev:/dev/urandom

# Log only a summary message on TLS handshake completion - no logging of client
# certificate trust-chain verification errors if client certificate
# verification is not required. With Postfix 2.8 and earlier, log the summary
# message, peer certificate summary information and unconditionally log

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

Fig.5.3.22 Postfix configuration

```
  GNU nano 6.2                                  /etc/dovecot/dovecot.conf *
mail_uid = 2000
mail_gid = 2000

# Assign uid to virtual users.
first_valid_uid = 2000
last_valid_uid = 2000

# Logging. Reference: http://wiki2.dovecot.org/Logging
#
# Use syslog
syslog_facility = local5

# Debug
#mail_debug = yes
#auth_verbose = yes
#auth_debug = yes
#auth_debug_passwords = yes

# Possible values: no, yes, plain, sha1.
# Set to 'yes' or 'plain', to output plaintext password (NOT RECOMMENDED).
#auth_verbose_passwords = no

# SSL: Global settings.
# Refer to wiki site for per protocol, ip, server name SSL settings:
# http://wiki2.dovecot.org/SSL/DovecotConfiguration
ssl_min_protocol = TLSv1.2
ssl = required
verbose_ssl = no
#ssl_ca = </path/to/ca
ssl_cert = </etc/letsencrypt/live/mail.rtrpcyber.xyz/fullchain.pem
ssl_key = </etc/letsencrypt/live/mail.rtrpcyber.xyz/privkey.pem
ssl_dh = </etc/ssl/dh2048_param.pem

# Fix 'The Logjam Attack'
ssl_cipher_list = EECDH+CHACHA20:EECDH+AESGCM:EDH+AESGCM:AES256+EECDH
ssl_prefer_server_ciphers = yes

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

Fig.5.3.23 Dovecot Configuration

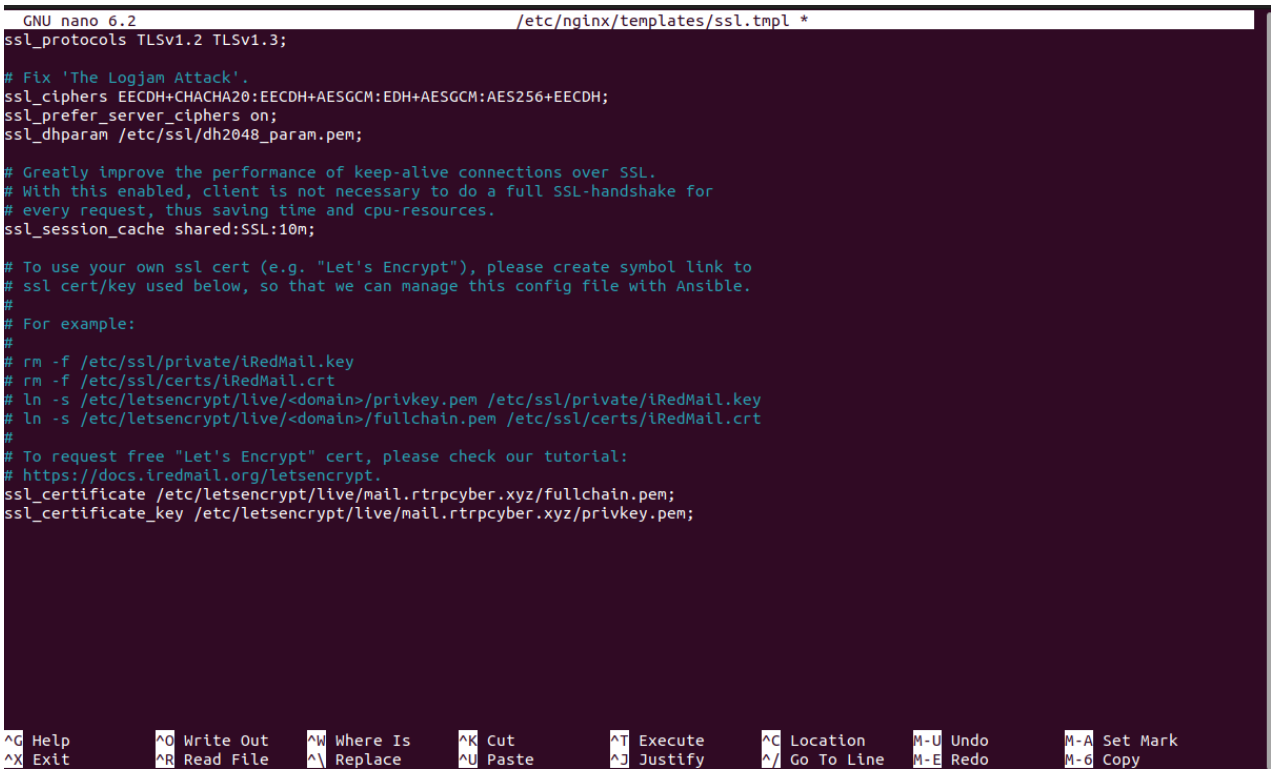

Fig.5.3.24 Nginx SSL Configuration

Fig.5.3.25 Reload NginX

```
cyberpunk@mail:~$ sudo amavisd-new showkeys
[sudo] password for cyberpunk:
; key#1 2048 bits, s=dkim, d=rtrpcyber.xyz, /var/lib/dkim/rtrpcyber.xyz.pem
dkim._domainkey.rtrpcyber.xyz.  3600 TXT (
  "v=DKIM1; p="
  "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAuA2Z18gij6t1KwetC1YO"
  "+Oj+vq0NpBSLFnVESx1jSBmaRKwdkhWq1pmc6TEmGVNSjpezxcdH1okNacfauehq"
  "SgH8fpPHD8UA5MuR4Fas+cE5hkljyBCZeogXNy/rtFNOoy4RqK4l+u65KonH1fMp"
  "Io6p5uQDLsgL4kh92fWSq1imUIQM2SSMi0iA6JOgf/F8XJ6c2LVatuUTvOJ2vNCx"
  "OB0r79IPFZd2b3vv+YKL8X57IxNMHuY+xRGswhVmhMvgx/GIsuIyAnIJcyX4z27i"
  "GCazdE8+uFAzLEaK1JEaT0v9FQBkMLtxtA9559lTKBKqCylo9Clt76mGkrdCv8b4"
  "nQIDAQAB")
```

Fig.5.3.26 DKIM Entry

```
cyberpunk@mail:~$ sudo swapon --show
[sudo] password for cyberpunk:
cyberpunk@mail:~$ sudo swapon --show
cyberpunk@mail:~$ sudo fallocate -l 1G /swapfile
cyberpunk@mail:~$ sudo chmod 600 /swapfile
cyberpunk@mail:~$ sudo mkswap /swapfile
mkswap: /swapfile: warning: wiping old swap signature.
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=721dfbe0-ef5c-4c4b-b99f-cc71cdcf11ac
cyberpunk@mail:~$ sudo swapon /swapfileclient_loop: send disconnect: Broken pipe
cyberpunk@cyberpunk-VirtualBox:~$ ssh cyberpunk@103.54.56.183
cyberpunk@103.54.56.183's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-107-generic x86_64)

 System information as of Thu Jun  6 07:59:32 AM EDT 2024

  System load:  0.0                Processes:             157
  Usage of /:   33.5% of 19.58GB   Users logged in:       0
  Memory usage: 54%                IPv4 address for eth0: 103.54.56.183
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Thu Jun  6 04:57:12 2024 from 106.208.13.143
cyberpunk@mail:~$ sudo swapon /swapfile
[sudo] password for cyberpunk:
cyberpunk@mail:~$ sudo nano /etc/fstab
cyberpunk@mail:~$ sudo systemctl daemon-reload
cyberpunk@mail:~$ sudo systemctl restart clamav-daemon
```

Fig.5.3.27 Swapping the files.

```
cyberpunk@mail:~$ sudo amavisd-new testkeys
[sudo] password for cyberpunk:
TESTING#1 rtrpcyber.xyz: dkim._domainkey.rtrpcyber.xyz => pass
cyberpunk@mail:~$
```

Fig.5.3.28 DKIM Entry Passed

Type	Host	Value	TTL
A Record	@	103.54.56.183	Automatic
A Record	mail	103.54.56.183	Automatic
TXT Record	@	v=spf1 mx ~all	5 min
TXT Record	_dmarc	v=DMARC1; p=none; pct=100; rua=mailto:dmarc@rtrpcyber.xyz	5 min
TXT Record	dkim._domainkey	v=DKIM1; p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgK...	5 min

Fig.5.3.29 TXT Records

DNSSEC ? Status

MAIL SETTINGS ? Custom MX

Actions

Type	Host	Value		TTL
MX Record	@	mail.rtrpcyber.xyz.	10	Automatic

Fig.5.3.30 MX Record

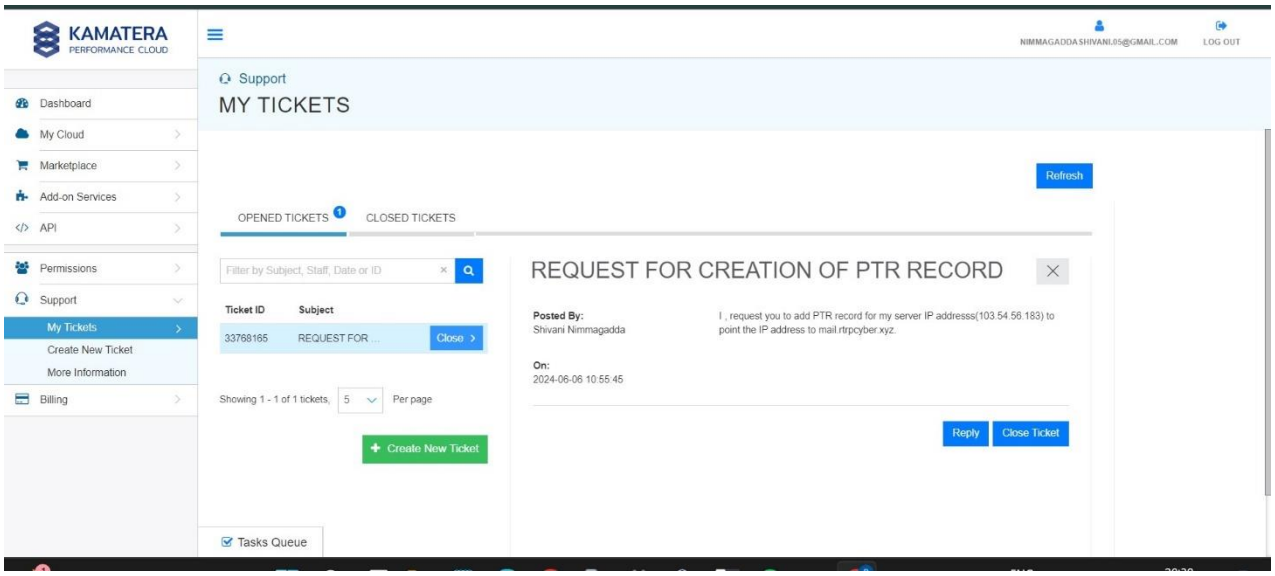

Fig.5.3.31 Request for PTR Record

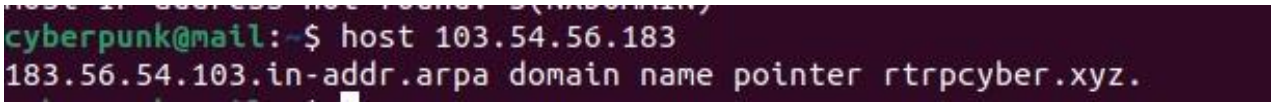

Fig.5.3.32 PTR Record

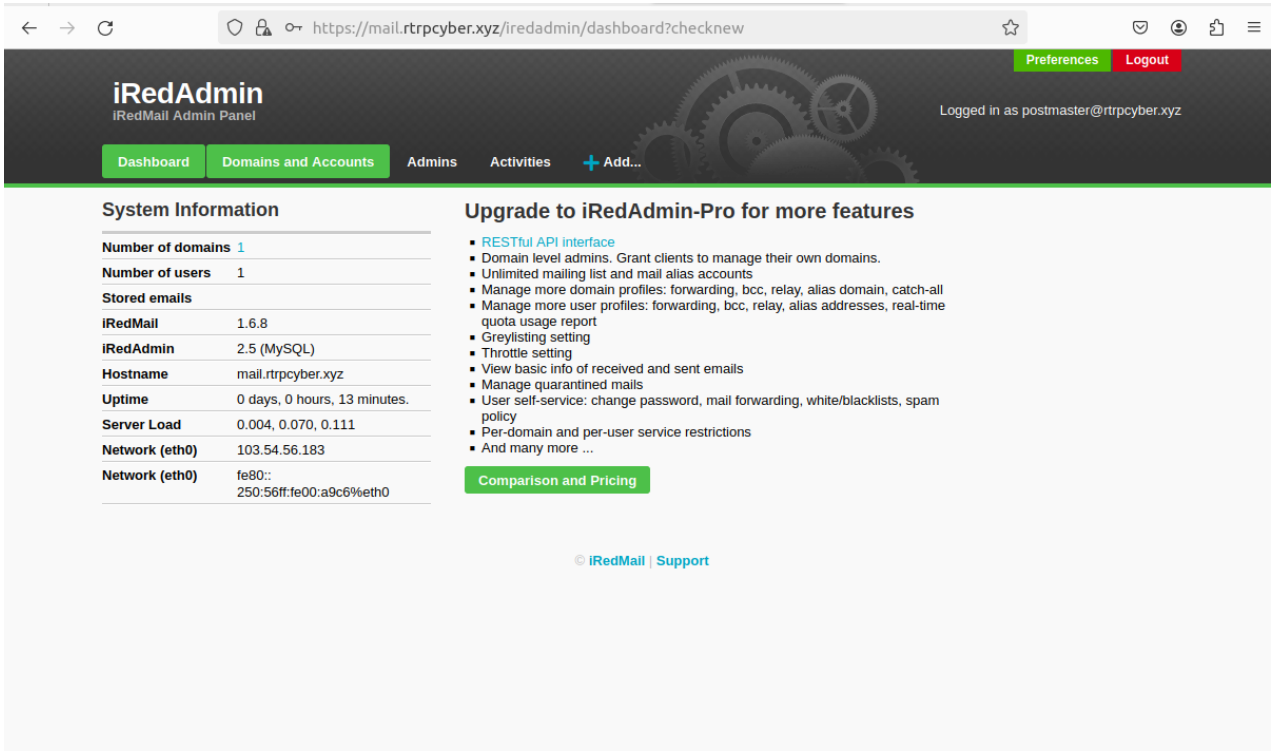

Fig.5.3.33 iRedAdmin Login Page

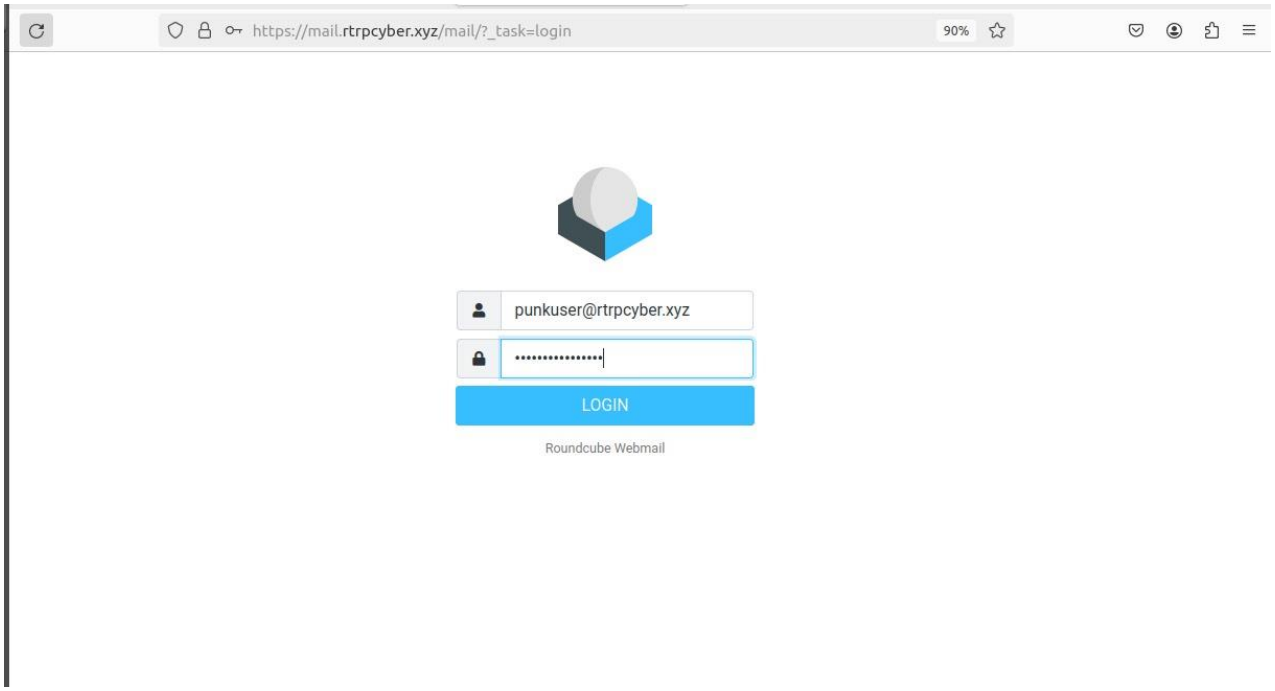

Fig.5.3.34 Roundcube Login Page

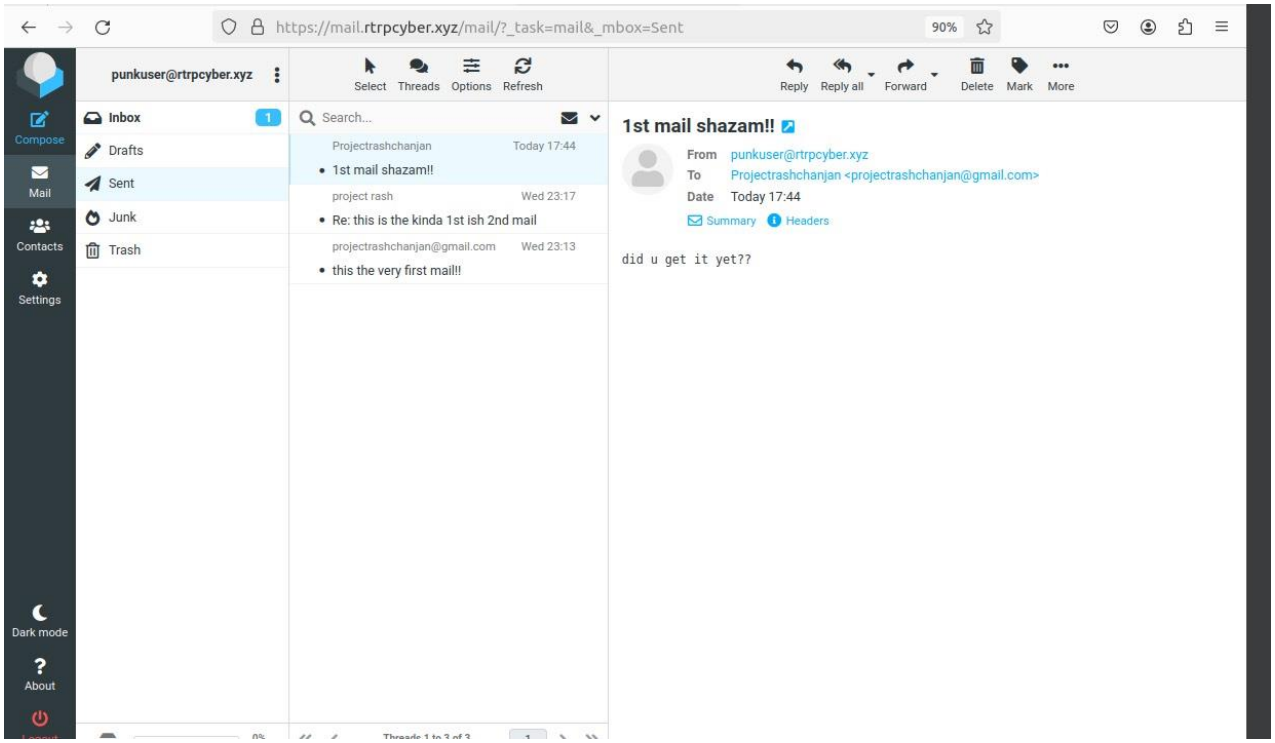

Fig.5.3.35 Roundcube

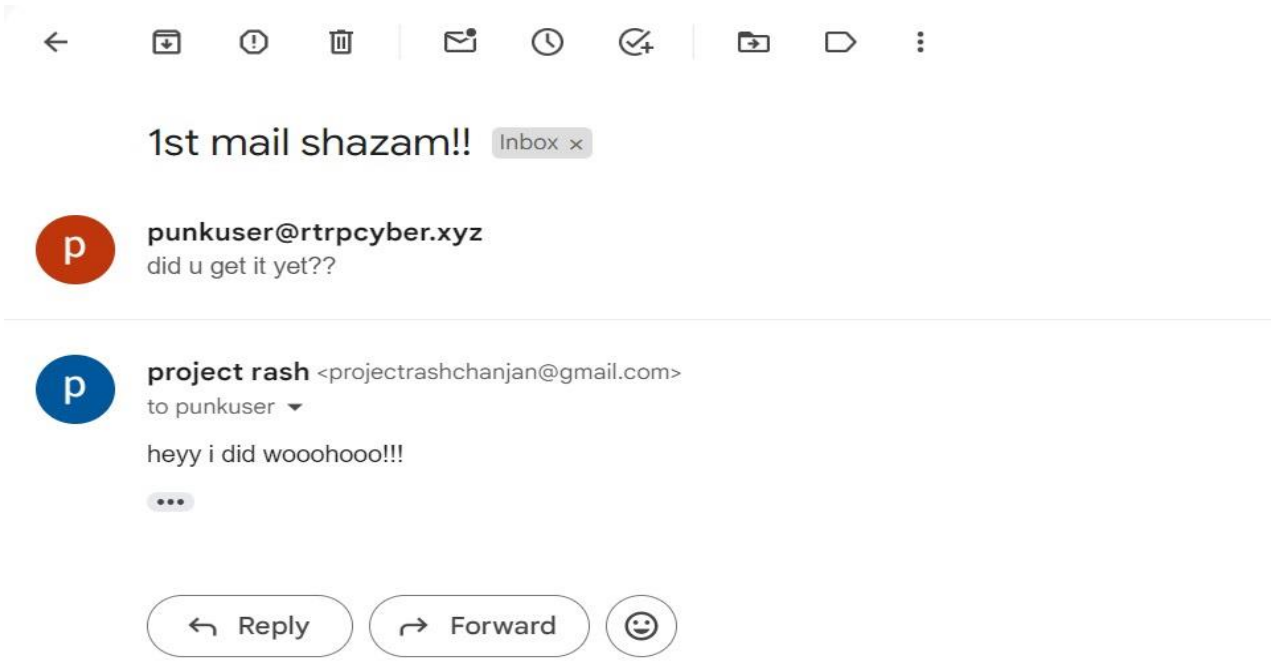

Fig.5.3.36 Mail successfully received

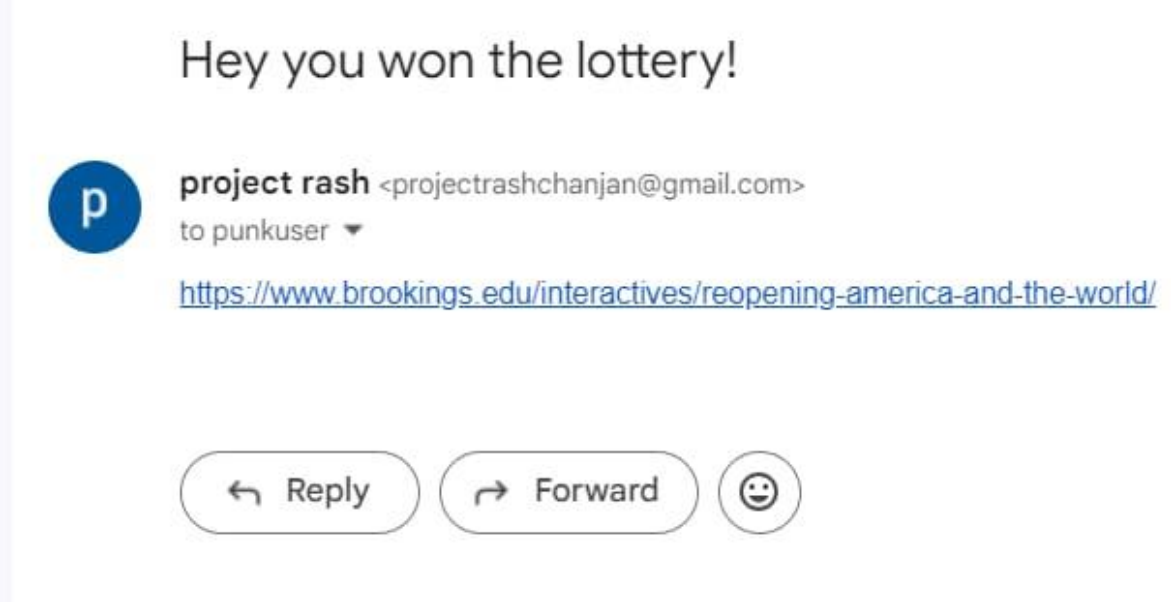

Fig.5.3.37 Mail sent containing malicious content

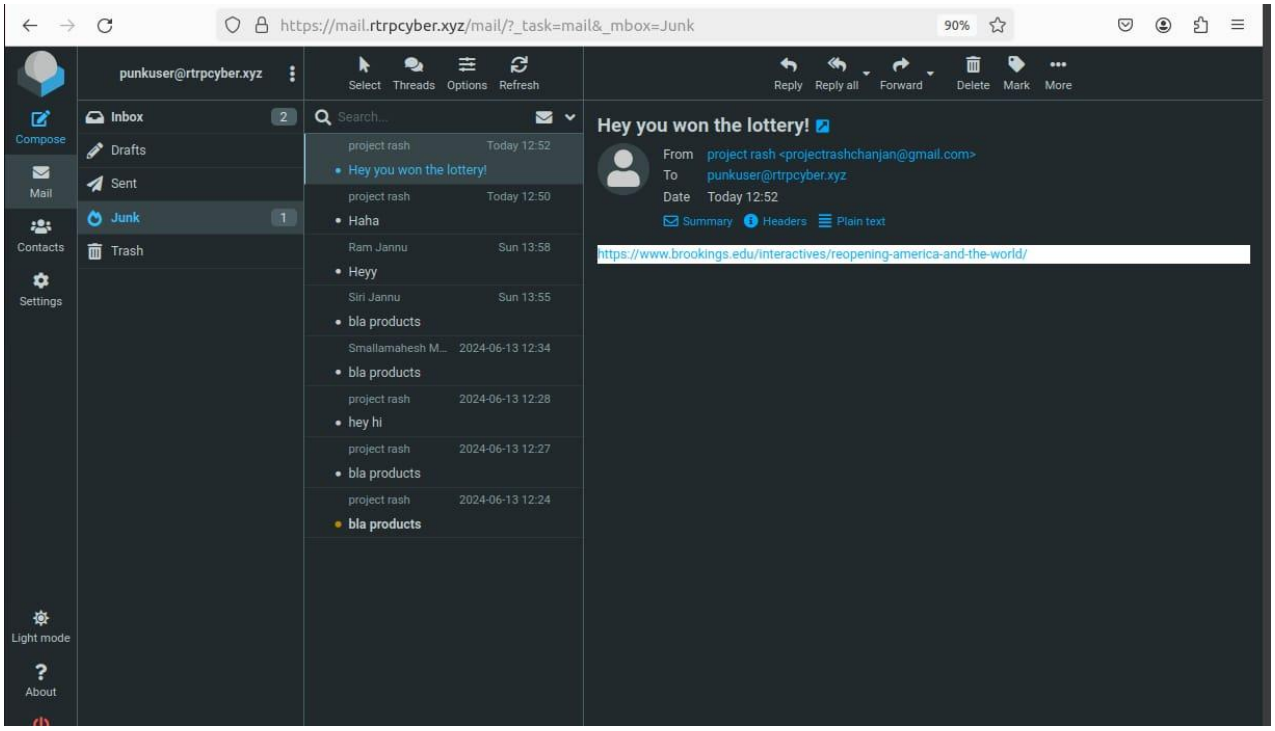

Fig.5.3.38 Mail blocked and stored in junk

5.4 Test Cases

Test Case ID	Test scenario	Test Case	Pre-conditions	Test Steps	Test Data	Expected Results	Post-Conditions	Actual Results	Status
1	Server Setup	Verify server installation and basic configuration	Server hardware available, OS installed	1. Install email server software 2.Configure basic settings	Server software package	Email server installed and running with basic configuration	Email server operational	The email server is opened and running, pointing to the correct IP address of the domain. Opens the web panel for user login.	PASS
2	Server Setup	Verify SSL/TLS configuration	Email server installed	1.Generate SSL certificate 2. Configure server to use SSL/TLS	SSL certificate details	Email server accepts connections over SSL/TLS only	Email server uses secure connections	The SSL certificate verifies the IP address (103.54.56.183) and DNS records (A, MX) are pointing to the correct domain, allowing the email server to accept connections	PASS

3	User Authentication	Verify user authentication	Email server installed and configured	1. Create user accounts 2. Attempt login with valid credentials	Username and password	Users can log in successfully with correct credentials	Users logged in successfully	User creates an account in iRedMail admin under the domain (rtrpcyber.xyz). User successfully logs into the email server with correct credentials.	PASS
4	User Authentication	Verify failed login attempt with incorrect credentials	User accounts created	1. Attempt login with invalid credentials	Invalid username and/or password	Login attempt fails, error message displayed	No unauthorized	1. Incorrect credentials entered by the user leads to pop-up “Login failed”. 2.User enters unregistered credentials, pop-up alert “User not found”.	PASS

5	Malware Protection	Verify malware detection on incoming emails	Email server installed, antivirus configured	1. Send an email with malware attachment	Email with malware attachment	Email server detects and blocks the email, sends alert	Email containing malware is blocked	Email is sent by external mail address. Email server(mail.rtrpcyber.xyz) detects the malicious content present in the mail and blocks the mail.	PASS
6	Verify malware detection on outgoing emails	Verify malware detection on outgoing emails	Email server installed, antivirus configured	1. Send an email with malware attachment	Email with malware attachment	Email server detects and blocks the email, sends alert	Email containing malware is blocked	Email is sent from the internal server. Email server detects the malicious content and blocks the mail.	PASS

7	Email Filtering	Verify spam filtering	Email server installed and configured with spam filters	1. Send spam email	Spam email content	Spam email is flagged and moved to spam/junk folder	Spam folder updated with new spam email	Email is sent to the user(punkuser@rtrpcyber.xyz). Email server detects the spam content in the body and subject, the email is flagged as spam and moved to junk folder.	PASS
8	Email Filtering	Verify email whitelist functionality	Email server installed and configured with whitelist	1. Add sender to whitelist 2. Send email from whitelisted sender	Whitelisted email address	Email from whitelisted sender bypasses spam filter	Email received in inbox	Email is received from the sender (projectraschanjan@gmail.com), since the email is not spam i.e. whitelisted. Email is successfully received	PASS
9	Backup and Recovery	Verify backup functionality	Email server installed and running	1. Configure backup 2. Trigger backup	Backup configuration settings	Email server data is backed up successfully	Backup files created and stored in specified location	Emails received and the data is stored and maintained in /var/log/mail-log directory.	PASS

10	Security and Access Control	Verify role-based access control	Email server installed and configured	1. Configure roles and permissions 2. Attempt access with different roles	Role-specific user accounts	Users can only perform actions according to their role's permissions	Access control enforced	Email server receives high volume of mails . Leads to server slowing down.	FAIL
11	Security and Access Control	Verify audit logging	Email server installed and running	1. Perform various actions 2. Check audit logs	Actions such as login, send/receive emails	All actions are logged correctly in audit logs	Audit logs contain detailed records of actions	Email server receives email with large amount of attachments , leads to reduce in server performance rate. Leads to more time consumption in processing the mail.	FAIL

Table 5.4: Test Cases

6.Conclusion and Future Scope

This Project keeps emails safe from viruses, benefiting everyone from individuals to governments, it builds trust in digital world by keeping emails secure. It's all about keeping emails real and safe from any kind of phishing attacks online. This project ensures to make sure that emails are totally and completely free from harm. Our project addresses the pressing need for a robust e-mail infrastructure in today's digitally vulnerable world. By employing an arsenal of tools and technologies, we not only establish an environment fortified against malware but also champion the cause of confidential and trustworthy electronic communication. The secure email servers with malware protection are much used in business**,** Financial Institutions**,** Government Agencies, Education etc.

As we are living in a world of online dangers and cyber threats, organizations might need a secure email server for themselves. This project could help them make an email system which can be very safe and secured. Finally, this project could also become easier to use and more efficient tool for everyone wanting trustworthy emails.

7.References

[1] Arun Vishwanth, Fredrik Heiding, Peter S. Park, Jeremy Bernstein, Bruce Schneier: Devising and Detecting Phishing Emails Using Large Language Models,11 March 2024.

[2] Shuji Sakuraba, Minami Yoda ,Yuichi Sei: Improvement of Legitimate Mail Server Detection Method using Sender Authentication,20-22 June 2021.

[3] Guter Kalem, Pinar Sarisaray Boluk: **:** The effect of social media user behaviors on security and privacy threats,30 May 2022.

[4] Eltigani, Abdelsatir: On the Implementation of a Secure Email System with ID-based Encryption,10 September 2020.

[5] Sanaa Kaddoura ,Omar Alfandi ,Nadia Dahmani:A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach, 01 February 2021.

[6] Shafiya ,Tariq Banday: Improving Efficiency of E-mail Classification Through On-Demand Spam Filtering,15 September 2020.

[7] Ramkumar Rajendran, Mohammod Naveed Aman, Biplap Sikdar Asif: Unveiling the connection between Malware and Pirated software in South East Asian countries,30 Jan 2024.

[8] Rasha Zieni, Maria Carla: Phishing or Not Phishing? A Survey on the Detection of Phishing Websites,22 February 2023.

[9] Abdullah Hussein AI-Ghushami, Dubeeruddin syed, Ameena Zainab, Haya Abdelshahid: Email Security concept, Formulation and Applications,13 January 2023.

[10] Shanta nu, Janet ,R Joshua Arul Kumar: Malicious URL Detection :A Comparative Study, 12 April 2021.

[11] Young-seob Jeong ,Sang-Mi Lee, Jong-Hyun Lee: Malware detection using Byte Streams of different File Formats,10 May 2022.

[12] Laxshamana Rao Kalabange ,Srinivasa Rao :A Boosting-Based Hybrid Feature selection and Multi-Layer Stacked Ensemble Learning Model To Detect Phishing Websites, 10 July 2023.

[13]Sanjay Adiwal, Akanksha Gupta, Balaji Rajendra, B S Bindhunmandhava:A Secure Methodology for Filtering Spam & Malware in E-Mail System and Secure E-mail Testbed Setup, 12 May 2021.

[14] Jian Gao , Fucai Zhou: An Encrypted Cloud Email Searching and Filtering Scheme Based on Hidden Policy Ciphertext-Policy Attribute-Based Encryption With Keyword Search., 6 December 2021.

[15] M. Francisca Hinarejos, Josep-Lluis Ferrer-Gomila:A Solution for Secure Multi-Party Certified Electronic Mail Using Blockchain.29 May 2020.

UBUNTU INSTALLATION ON VIRTUAL MACHINE

1) **Install and configure VirtualBox:**

Go to the official website of VirtualBox and download the latest version of VirtualBox form the available packages.

2) **Download Ubuntu desktop ISO:**

Download the latest ubuntu version for your windows and save it on your pc.

3) **Install Ubuntu on VM:**

Open your VirtualBox and create a new virtual machine on it. Add a name to your virtual machine and load the ubuntu ISO file you previously downloaded from the ubuntu website.

Create a user profile and allocate appropriate resources like CPU, memory and storage to the virtual machine. Finally click finish to initialize the machine.

4) **Start the VM:**

Click start button to launch the virtual machine. Once the installation completes, you will be asked. to enter the password to login. Once you are logged in, new Ubuntu desktop is appeared.

Initial step after installing ubuntu is to run the update command in terminal, by running the update command everything gets updated to the latest versions.