National University of Computer & Emerging Sciences
Karachi Campus

PROJECT PROPOSAL

Course Name: Computer Networks (CN)

Course Instructor: Muhammad Usman

"Implementation of SMTP Server"

Group members:

Name	Roll No	Class
Maisum Abbas (Leader)	22K-4129	BAI-6A
Mudasir	22K-8732	BAI-6A

Project Proposal:

Introduction

In today's interconnected world, email communication is an essential tool for businesses, educational institutions, and individuals. A properly configured SMTP (Simple Mail Transfer Protocol) server enables users to send and receive emails efficiently and securely. This project aims to implement a custom SMTP mail server using Postfix and Sendmail to handle email transmissions while ensuring secure authentication and secure protocols.

Project Description

Overview

This project involves setting up and configuring an SMTP mail server that enables secure email communication over a network. The server will be built using Postfix and Sendmail with support for user authentication, encrypted email transmission, and activity logging. The system will provide a basic command-line email interface using Python’s smtplib and can be expanded to include a Flask-based web interface for ease of use.

How It Works

1. ### SMTP Server Setup

   Install and configure Postfix and Sendmail.

   Define email routing and forwarding rules to ensure efficient mail delivery.

2. ### Secure Email Configuration

   Enable TLS (Transport Layer Security) encryption to protect email contents from interception.

3. ### Email Sending and Receiving Functionalities

   Users can send emails via Python's smtplib module.

   Email logs will be maintained to track email activity, including sender, recipient, and timestamps.

4. ### Web-Based Email Interface for both sender and Receiver

   A Flask-based web interface may be developed to allow users to send and receive emails through a browser.

   The frontend will communicate with the SMTP server to facilitate email exchange.

Working Plane & Group Members Contributions (5 Weeks)

Week#1: Simple Mail Transfer Protocol (SMTP) Server Setup

- Install and configure Postfix and Sendmail for email handling. **(Group Member 1)**
- Verify basic email sending and receiving functionality. **(Group Member 2)**

Week#2: Secure Communication Implementation

- Enable TLS encryption for email security. **(Group Member 1)**
- Configure user authentication to ensure only authorized users can send emails. **(Group Member 2)**

Week#3: Web-Based Email Interface

- Develop a Flask-based frontend for easier email management. **(Group Member 1)**
- Integrate the web interface with the SMTP server. **(Group Member 2)**

Week#4: Testing & Security Enhancements

- Conduct unit testing to verify correct email transmission. and Implement logging and monitoring mechanisms to track email activity and potential issues**.(Group Member 1)**

Week#5: Final Integration & Report Submission

- Perform final security tests and optimizations. **(Group Member 1 & 2)**
- Prepare and submit the final project report and documentation. **(Group Member 2)**

References

Official Documentation & Tutorials:

- Postfix & Sendmail Official Documentation: https://www.postfix.org/
- SMTP Protocol Specification: https://tools.ietf.org/html/rfc5321
- TLS Encryption for SMTP: https://www.openssl.org/
- Email Authentication Standards: https://dmarc.org/

Python Libraries & Modules:

- Python smtplib for sending emails: https://docs.python.org/3/library/smtplib.html
- Python email module for composing messages: https://docs.python.org/3/library/email.html
- Python Flask for web-based interface: https://flask.palletsprojects.com/

Security & Networking References:

- Secure Email Transmission & TLS: https://www.ietf.org/rfc/rfc3207.txt