Trust-Enhanced SMTP Protocol Simulation for Secure Email Transmission

CSE 4106 Computer Network Laboratory

Submitted to:

Md. Sakhawat Hossain
Lecturer
Department of CSE, KUET

Farhan Sadaf
Lecturer
Department of CSE, KUET

Submitted by:

Peyal Saha
Roll: 2007001
Dept. of CSE, KUET

1 Objectives

- To simulate the complete working process of the SMTP email protocol using OMNeT++.
- To implement SMTP Client, Server and Trust Manager modules that communicate over a network.
- To design a Trust Manager that filters or forwards emails based on the sender's trust score.
- To analyze how different trust levels affect message delivery, dropping and forwarding.
- To visualize end-to-end message flow and trust-based filtering through OMNeT++ animation.
- To collect and evaluate statistics such as total emails sent, dropped and successfully delivered.

2 Introduction

The protocol used for sending, receiving and relaying email messages from one server to another is known as SMTP. For this project, we set up a simulation model of the SMTP process using OMNeT++. OMNeT++ is a discrete event simulation framework that is primarily used for modelling of communication networks. The simulation is a setup of two email hosts. One is the sender of the message, and the other is the receiver of the message. Also, there are two mail servers. These servers are used to send a message and receive a message.

To improve the authenticity of the model, Trust Manager module is added amidst the mail servers. The Trust Manager works like a filter firewall that evaluates the trust score of every email. The decision to accept or reject a message is made based on its score. If the trust score of an email is less than a preset threshold value, it is dropped. This simulates the dropping of spam and malicious emails. If the trust score is acceptable, it is forwarded for normal delivery.

According to the simulation, the use of trust-based filtering in communication networks such as email networks can reduce spam. By analyzing the system's behavior under different trust threshold settings, we can better understand the impact of trust evaluation on overall network performance and message reliability.

3 Motivation

- In real-world email systems, spam and phishing attacks are common and need effective filtering.
- Untrusted or low-reputation senders often deliver harmful or unwanted messages to users.
- A trust-based mechanism helps identify reliable senders and reduce the spread of spam.
- Simulating this concept in OMNeT++ allows clear visualization of how trust affects message delivery.
- The project provides an educational approach to understand email security at the network layer.
- It bridges theoretical knowledge of SMTP with practical trust management and filtering techniques.

4 Theory and Methodology

Theory

The Simple Mail Transfer Protocol (SMTP) is the standard protocol used for sending email messages across networks. It defines how an email client communicates with a mail server, and how mail servers exchange messages with one another.

The communication begins with the client initiating a connection to the mail server. Once connected, the client sends a sequence of commands such as:

- `HELO/EHLO` – to identify the client to the server.
- `MAIL FROM:` – to specify the sender's address.
- `RCPT TO:` – to specify the recipient's address.
- `DATA` – to send the actual message body.
- `QUIT` – to close the connection.

For every command, the server sends a response code to indicate the result:

- `220` – Service ready.
- `250` – Requested action completed successfully.
- `354` – Start mail input.

- `550` – Action not taken (e.g., message rejected).

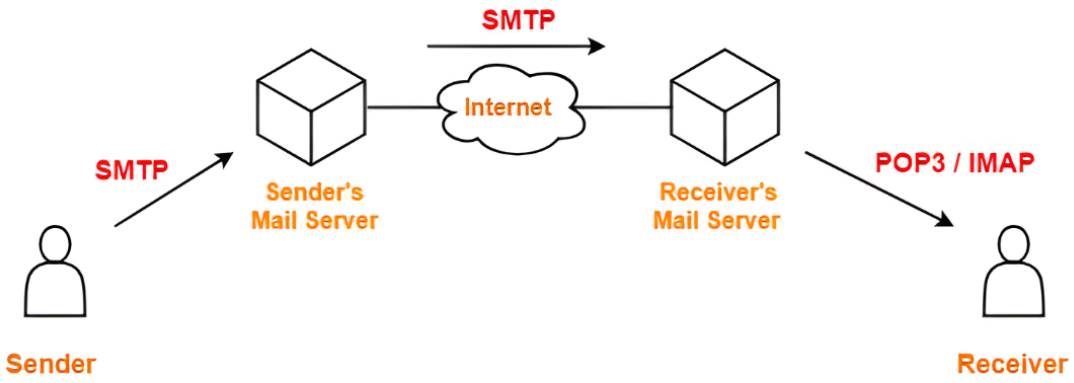

Figure 1: SMTP Protocol Message Exchange between Client and Server

This simulation extends the traditional SMTP protocol by introducing a **trust-based filtering mechanism**. Each email message carries a *trust score* between 0.0 and 1.0, representing how reliable or safe the sender is. The Trust Manager uses this score to decide whether to forward or drop the message.

- If the trust score is **below** the threshold (e.g., 0.3), the message is considered untrusted and is dropped.
- If the score is **above** the threshold, the message is allowed to pass through to the next mail server.

Methodology

The simulation consists of several interconnected modules developed in OMNeT++:

1. **SMTPClient:** Generates emails, assigns trust scores, and starts the SMTP session by sending commands.
2. **SMTPServer:** Receives client commands, sends appropriate response codes, and handles message transfers.
3. **TrustManager:** Acts as a filter or gateway that checks the trust score before forwarding or dropping messages.
4. **SimpleChannel:** Simulates the physical link with specified data rate and propagation delay.
5. **TrustSMTPNetwork:** Connects all components to form the full communication path.

There are two main communication scenarios:

- **High Trust (HostA → HostB):** Messages from HostA have a high trust score (0.9) and are successfully delivered through all servers.
- **Low Trust (HostB → HostA):** Messages from HostB have a low trust score (0.2), which causes the Trust Manager to block them before reaching the destination.

HostA → ServerA → TrustManager → ServerB → HostB

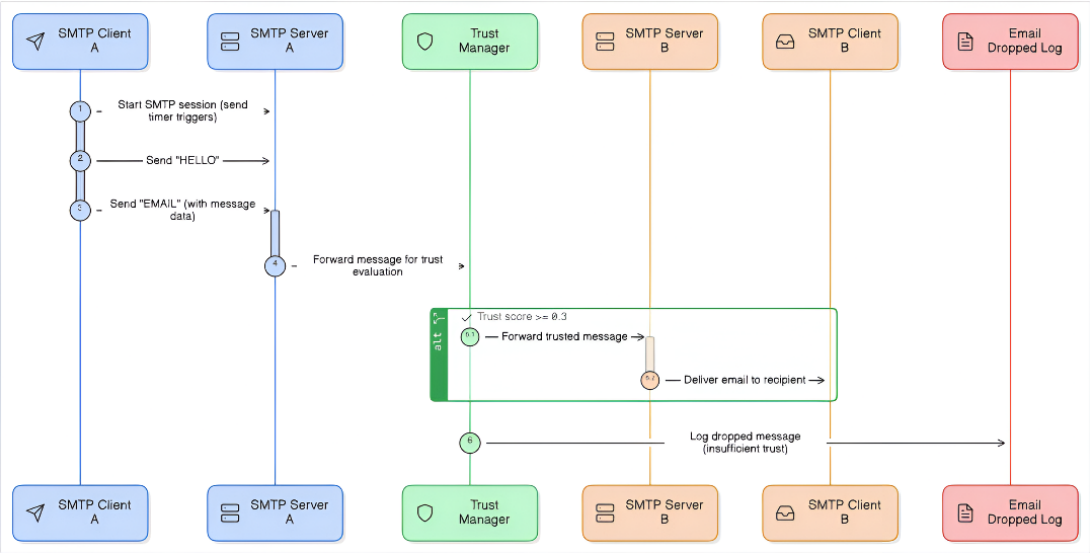

Figure 2: Flowchart of Trust-Aware SMTP Simulation Process

5 Discussion

The OMNeT++ event log, as well as the animation, were used during the simulation to observe the behavior of the system. Each message sent by the clients was found to come with a trust score. The score was checked against a threshold value when the message got passed through the Trust Manager. If the trust score exceeded the threshold, the message moved on to the subsequent mail server and finally reached the host destination. Conversely, the Trust Manager immediately dropped messages that scored lower than the threshold.

Selective forwarding exhibited how the trust mechanism impacted communication impact in total. For instance, HostA with a score of 0.9 was able to seamlessly send and receive messages. In contrast, HostB with a score of 0.2 could not deliver messages because they were blocked. EMANE Event Log entries in OMNeT++ confirmed the detail of forwarding and dropping actions depending upon the rules.

In addition, the simulation also showed how SMTP commands like HELO, MAIL FROM, RCPT TO, and DATA are sent and received by the client and server. The Trust Manager was effective at blocking off packets based on reputation rather than by the content.The overall result showed a balanced integration of protocol behavior and trust-based decision-making. The graphical interface of OMNeT++ made it easier to understand the dynamic process and the role of each component in the network.

6 Conclusion

The Trust-Aware SMTP simulation demonstrated that a better email security can be achieved by adding a trust based layer. This project combined the basic workings of the SMTP protocol with a trust management mechanism that is relatively simple. The result showed that messages from trusted sources are delivered without a hitch, while messages from untrusted or suspicious sources are automatically blocked before delivery.

The sender reputation of emails helps in identifying spam and phishing. This is also what we see in real life. Running the simulation made the processing, filtering and logging of messages in a network environment very clear. The project helped to better understand SMTP message flow. It also demonstrated how trust evaluation can be incorporated into communication protocols. In general, this simulation gives a user-friendly way to study secure and trustworthy data transmission in computer networks.

7 References

1. GeeksforGeeks, ”Simple Mail Transfer Protocol (SMTP)”, https://www.geeksforgeeks.org/computer-networks/simple-mail-transfer-protocol-smtp/

2. Cloudflare Learning, ”What is SMTP”, https://www.cloudflare.com/learning/email-security/what-is-smtp/

3. OMNeT++ Tutorials, ”Getting Started with OMNeT++”, https://docs.omnetpp.org/tutorials/tictoc/part1/

4. TutorialsPoint, ”SMTP Protocol”, https://www.tutorialspoint.com/computer_fundamentals/computer_networks_smtp.htm

5. Guru99, ”SMTP in Computer Networking”, https://www.guru99.com/smtp-simple-mail-transfer-html