Ciências ULisboa Faculdade de Ciências da Universidade de Lisboa

Final Project
Data Privacy and Security

Inês Ferrejal 62567
João Garcia 62884
Ezequiel Barreira 44768

8/12/2023

Contents

1 Overview

This project consists of two main programs: the Server and the Client. The Server maintains a list of online users, while the Client communicates with the Server and enables users to exchange messages securely.
The Server program remains online and provides information about online users to the Client. Users are added to the $registered_users.csv$ file when they appear online for the first time, including their public key and IP:port information. Once a user is online (and consequently in the registered users file), another client can send messages to them. The message exchange process occurs directly through an open socket connection.

2 Programs

2.1 Server

The Server program is responsible for:

- Maintaining a list of online users.
- Receiving a user's public key, IP address (with port) and Username (which this last can be edited each time the user enters the app).
- Expecting a ping from the Client every 10 seconds, if the ping is not received, the user is removed from the online list.
- Providing the list of online users, including their public keys, IP addresses, and usernames.

2.2 Client

The Client program periodically pings the Server to indicate that it is online and its responsible for:

- Communicating with the server using sockets
- Maintaining a list of online users by reading the Server's responses.
- Sending messages to online users by specifying their username.

3 GUI

The project initially started with a console-based interaction for testing purposes. However, to improve the readability and user experience, a Graphical User Interface (GUI) was later implemented.
To facilitate this implementation we used Swing, which enhanced the usability of the application. The GUI includes a chat panel that is initially invisible

and becomes visible once a user registers. The registration process involves entering a username in a text field and clicking a button. After the application loads it detects if the user is already registered by checking if the IP address of the current user is in the $registered_users.csv$ file, if it is, then the username is updated, if not, the new user is added to the file. After this initial check, registered users are loaded into a chat panel, and the user can interact with it to send messages.

4 Uses

This application serves as a secure peer-to-peer messaging platform. The application uses end-to-end encryption and digital signatures to protect the messages. Users can see who is registered and send them messages directly.

4.1 Dependencies

This project relied on the Java programming language for its implementation. Java's networking capabilities are used to establish socket connections between the server and clients, enabling real-time communication. Additionally Java's security features are also leveraged in this project. The use Java's cryptography libraries for end-to-end encryption and digital signatures allow for secure message exchange.

5 Security Considerations

This project is designed with a strong focus on security and privacy. It provides end-to-end encryption, ensuring that messages are only readable by the intended recipients. To further enhance security, messages are authenticated using digital signatures, which verify the sender's identity and ensure the integrity of the message. It is done through a RSA encryption with PSS padding for message security. Also, it's as decentralized as possible, the server only stores the online users, dealing with the registers and sending pings to see if they are online, and each client can generate its own public-private key and send encrypted messages directly to other clients without the need of a central server.

6 Limitations

- The messages are not persistent as they are not stored anywhere, ideally it should be stored in a mongoDB (p.e.) and do a searchable encryption on it to see the messages and make privacy and secure queries to retrieve it.
- On the GUI, the panel with the users show all the registered users, and if we try to send message to one that is not online, he will obviously not receive it, again because the messages are not persistent.

7 Final Comments

This project contributes to the field of Data Privacy and Security and is a practical demonstration of the concepts learned during the semester. By implementing P2P communication and secure it regarding privacy, integrity, authenticity, etc, we open ways to a better understanding of Data Privacy and to the development of more complex tasks.

8 References

- Slides of the course object
- `https://docs.oracle.com/javase/tutorial/uiswing/`