An End-to-end Encrypted Application

Xiangyu Gao, Rui Li, *Student,UW ECE*

***Abstract*—We designed and improved an end-to-end conversational application to enable facial recognition login, encrypted sessions, and other features to raise awareness about product design for security engineering.**

I. INTRODUCTION

We designed an end-to-end encrypted session system, which includes some functions, such as encrypted session, face recognition login, sensitive content recognition, user firewall function, etc., to comprehensively realize the privacy protection of users. The system prevents people from hacking into other people's sessions and stealing content. The system adopts a high-intensity face recognition function to ensure that user privacy is protected. And the session content is encrypted during transmission, so even if the content is stolen, the user's session message cannot be obtained intuitively.

II. TECHNICAL ATTACK

A. Potential Attack

Due to the concurrency and non-contact characteristics of face recognition, people can take advantage of it. In other words, if someone can have your high-definition photo, then he may also be authenticated by face recognition. We believe that most people only have one camera on their laptop - and few people carry a second camera with them, so using 2D face recognition is more pervasive. 2D-based face recognition uses 2D cameras to capture plane images. The research time is relatively long and it is used in many fields. However, due to the limitation of depth data loss in 2D information, the collected information is limited and the security level is not high enough.

In addition, hackers may also steal user-session information by hacking into the system and obtaining this information for other purposes.

B. Mitigate the attack

Our solution is to protect the privacy of users by designing the system, mainly by implementing face recognition and encrypted transmission. We avoided this by increasing the strength of the facial recognition, however - due to 2D recognition - this vulnerability cannot be completely avoided, it can be fooled by high-definition photos, we just reduce the probability of this happening.

III. SYSTEM OVERVIEW

We mainly implement projects from the following aspects: encryption algorithm, face recognition, firewall, sensitive data recognition, etc.

A. Overall System Logic Design

Our system like figure 1. The server side opens the connection, and the server side is kept in a standby state by cyclic calls. The client accesses and establishes a connection. Then perform facial recognition authentication, the server sends a request, and after the client takes a photo, the image is transmitted to the server in binary form for verification. If the verification is passed, enter the user login. If the user triggers the firewall policy, the server will alert you.

After logging into a session, encryption and decryption are performed as information is transmitted within the session. If there is sensitive information in the information, the system will automatically capture it and prompt it on the server-side. Finally, one of the servers or client-side proposes to terminate the dialogue, which will end the dialogue, the client-side will close the connection, and the server-side will continue to cycle, that is, the standby state.

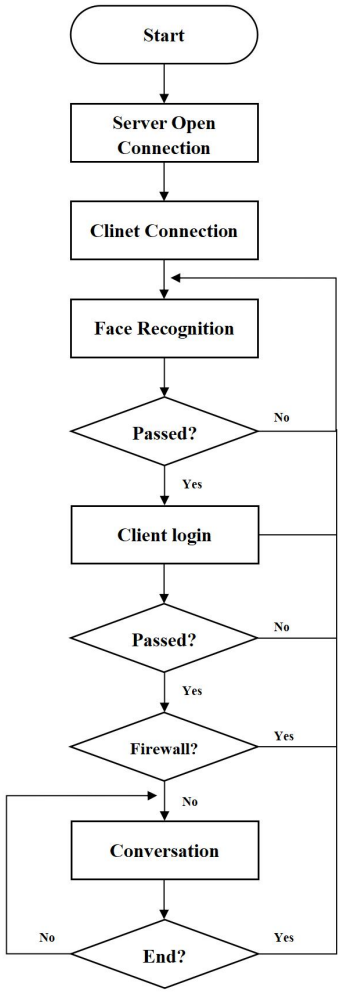

Figure 1. Flow chart

B. Encryption Algorithm

Here we are using asymmetric encryption.key1 and key2 are a pair of keys that are very different and hard to guess from one to the other. We can publicly transmit it to the left side Key1, and then the left side calculates the ciphertext and transmits it to the right side, and the right side decrypts it with Key2. The encrypted transmission method is as follows.

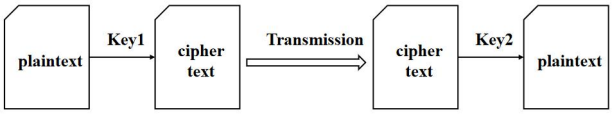

Figure 2. Encrypted transmission method

C. Face Recognition

In the face recognition part, the requests library, OpenCV, and face++ document API are mainly used to judge whether it is the same person by judging the similarity between the image captured by the camera and the sample image. First, call the function to open the camera and display the image. After receiving the photographing command, capture the picture and convert it into a grayscale image and store it in the relative path. Then call the Face++ API, upload the sample image uploaded in advance (that is, the user's face image) and the image captured by the camera to Face++ for processing, and get the face tokens of the two images. Then write a judgment function to compare the face tokens of the two pictures and return the similarity of the two pictures. Finally, add a judgment statement and call the judgment function to judge the similarity of the two pictures. If the similarity is greater than 95%, it is considered to be the same person, and you can enter the system for the next step, that is, enter the user name and password to log in.

D. Firewall

Firewalls isolate risk areas from safe areas and set a block list to restrict users who are on that list. In general, the firewall we designed has the following functions:

- Restrict unauthorized users from accessing the server, and filter out unsafe services and illegal users.
- Prevent intruders from approaching the defense facilities of the internal network, and detect and alarm network attacks.
- Restrict internal users from accessing special sites.
- Record the content and activities of information passing through the firewall.

D. Sensitive Data Identification

During information transmission, if sensitive information appears in the information, the server detects the information and issues an alert.

IV. WORK DESCRIPTION

We wrote a python security project, including four py files, and defined three classes and dozens of functions. In total, more than 400 lines of code were written. When thinking about system design, we looked at more than a dozen papers and designed several drafts of running logic. Although the process is continuous thinking and rewriting, after unremitting efforts, we finally get a design that we think is relatively reasonable.

The following figure 3 is the result of the server, and figure 4 is the result of the client.

```
Waiting For Host To Connect......
Accept connection from ('127.0.0.1', 65421)
Picture Name Received
Picture Receiving...
Picture Received
Face loading...

Proofreading Faces...

Comparing Faces...

Verification Succeeded!
Client with IP 127.0.0.1 is connected, port is 65425
Client Login Successfully:
username:user2
password:112233
Mon Jun  6 12:05:24 2022
Hello World!
----->end_talk
Client Has Been Disconnected!
Waiting For New Host to Connect......
```

Figure 3. Server Progress

```
Name Sending...
Shoot or upload?
(Please input "shoot" or "upload"):shoot
Name Sent!

Picture Sending...
C:/UW/22 Spring/595/pj2/client_picture/client.jpg send over...
Picture sent
Face Recognizing...
Face Verification Succeeded!

                        --------  Welcome to 595 Serving System  --------

please enter user name: user2
Please enter password: 112233
Landed successfully!
----->Hello World!
Disconnected from server! !!

Process finished with exit code 0
```

Figure 4. Client Progress

V. OVERVIEW

In this course design, during the preparation period, we consulted some papers and blogs, as well as some learning resources on the Internet, read textbooks, and successfully reproduced the code after understanding the principles. After a thorough discussion, it was decided to implement the idea together rather than in pieces. This ensures the overall coherence and integrity of the system. Our group is actively involved, meeting regularly for brainstorming, and three to four times a week for the last month to code together. This will not only ensure that we communicate fully, but also ensure the efficiency of writing. We continued to debug the code, add features, and finally completed this project. During the process, we felt the charm of the safety project and were deeply attracted. We are more aware of and appreciate the importance of group cooperation, this is pleasant and impressive cooperation.

VI. CONCLUSION

We implemented an end-to-end encrypted session system, and implemented encryption algorithms, face recognition authentication, sensitive data capture, and user firewall functions in the system. In the process of project practice, I also felt the importance of security and privacy. Developers

must not only prevent system errors but also prevent hacker intrusion, which is a difficult problem. In the next step, we will continue to try to attack the system we designed and improve the system.

REFERENCES

[1] Zhao, Ran, Q., Yuan, L., Chi, Y., & Ma, J. (2015). Key Distribution and Changing Key Cryptosystem Based on Phase Retrieval Algorithm and RSA Public-Key Algorithm. Mathematical Problems in Engineering, 2015, 1 – 12. https://doi.org/10.1155/2015/732609

[2] Reddy. (2022). RM- RSA algorithm. Journal of Discrete Mathematical Sciences & Cryptography, 25(1), 1 – 13. https://doi.org/10.1080/09720529.2020.1734292

[3] Guggenmos, Häckel, B., Ollig, P., & Stahl, B. (2022). Security First, Security by Design, or Security Pragmatism – Strategic Roles of IT Security in Digitalization Projects. Computers & Security, 118. https://doi.org/10.1016/j.cose.2022.102747