# Program Documentation

## Program Design

### Overview

The program is designed to implement a simple authentication system using a client-server architecture. The components of this system include the Authentication Server (AS), Ticket Granting Server (TGS), Service Server (V), and the client. The primary goal is to provide secure access to a service server by authenticating clients through a multi-step process.

### Components

- \*\*Authentication Server (AS):\*\* Responsible for authenticating clients and providing them with a Ticket Granting Ticket (TGT).

- \*\*Ticket Granting Server (TGS):\*\* Provides service tickets to clients based on valid TGTs.

- \*\*Service Server (V):\*\* The server providing the requested service to authenticated clients.

- \*\*Client:\*\* The end-user or entity seeking access to the service.

### Data Flow

The data flows in the following sequence:

1. The client sends an authentication request to the AS.

2. The AS validates the client's identity and issues a TGT.

3. The client requests a service ticket from the TGS.

4. The TGS issues a service ticket, allowing the client to access the service server.

5. The client contacts the service server using the service ticket.

6. The service server responds to the client's request.

## Challenges and Solutions

### Key Generation

\*\*Challenge:\*\* Generating secure 8-byte and 16-byte keys.

### Encryption and Decryption

\*\*Challenge:\*\* Ensuring correct encryption and decryption of messages.

\*\*Solution:\*\* Used the pyDes library with the correct key and encryption modes (ECB) for data protection.

### Communication

\*\*Challenge:\*\* Establishing secure communication between the client, AS, TGS, and V.

\*\*Solution:\*\* Implemented socket communication to send and receive encrypted messages.

### Error Handling

\*\*Challenge:\*\* Handling exceptions or errors during the authentication process.

\*\*Solution:\*\* Implemented try-catch for that

## Lessons Learned

- Enhanced understanding of authentication and encryption protocols.

- Improved problem-solving skills, especially when dealing with encryption and secure communication.

- Gained knowledge of key management and the importance of secure key generation.

## Improvements

- Implement more advanced encryption and security features.

- Expand the functionality of the service server to offer more services.

## Conclusion

This program successfully implements a basic authentication system using a client-server architecture. It showcases a working knowledge of authentication, encryption, and secure communication. The multi-step process ensures that only authorized clients gain access to the service.

Test Screenshot:

Initial:  
文本

描述已自动生成

文本

描述已自动生成

文本

描述已自动生成

文本

描述已自动生成

Changed value(change the validation and lifetime to 120, the error is Invalid TGT. Authentication terminated. Did not affect others)

文本

描述已自动生成 文本

描述已自动生成

文本

描述已自动生成 文本

描述已自动生成