WEB CHAT APPLICATION

Presented By:
Bhuwan Paneru
Sandip Ghimire

AGENDA

1 Introduction

2 Problem Statement

3 Objective

4 Literature Review

5 Methodology

6 Result

7 Conclusion

8 Demo

INTRODUCTION

Chat applications are a type of messaging service that allows users to communicate in real-time, either through text messages, voice calls, or video calls.

Examples:

- **WhatsApp**
- **Facebook Messenger**
- **Telegram**

PROBLEM STATEMENT

It includes:

- **Realtime Communication**
- **Secure Communication**

OBJECTIVE

1. **Designing Web-Based Chat Application for Seamless Communication**

2. **User-Friendly Interface Design**

3. **Secure Chat with End-to-End Encryption**

LITERATURE REVIEW

Study of Existing System

Telegram

Transport Layer Security with MTProto

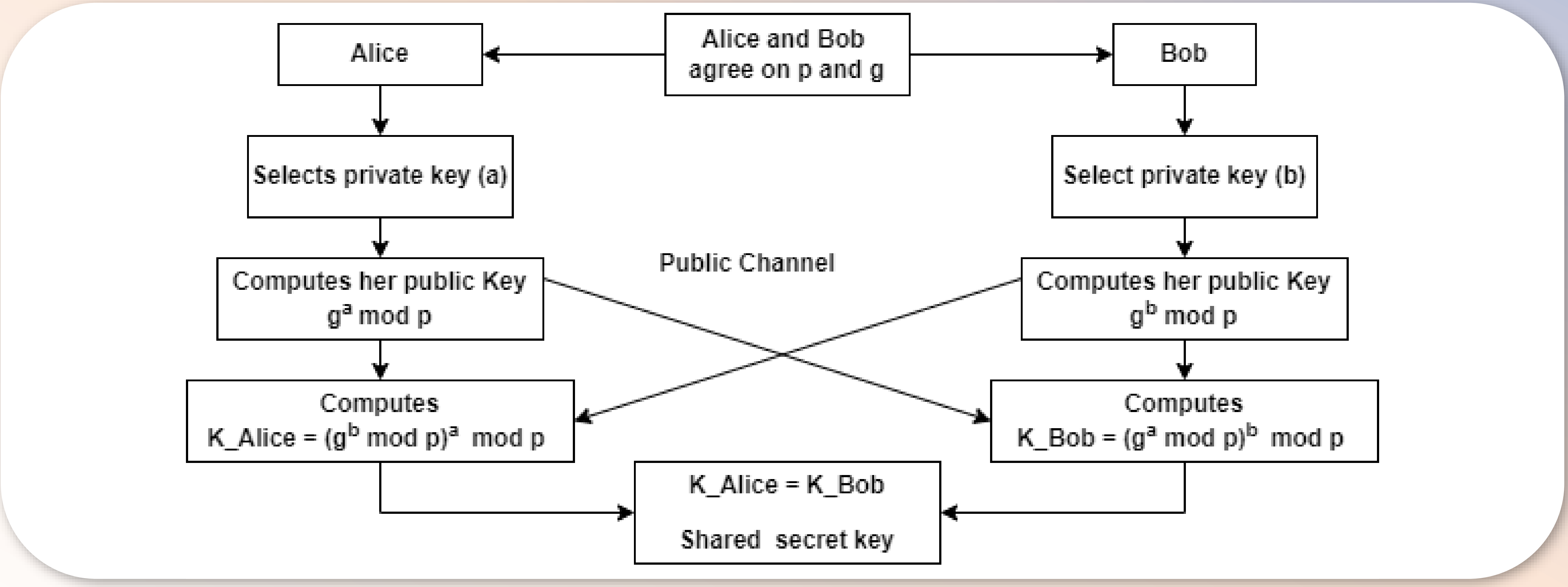
End-to-End Encryption of Secret Chats with AES and DHKE

Viber

Viber Encryption Protocol, a combination of Symmetric and Asymmetric Encryption algorithm

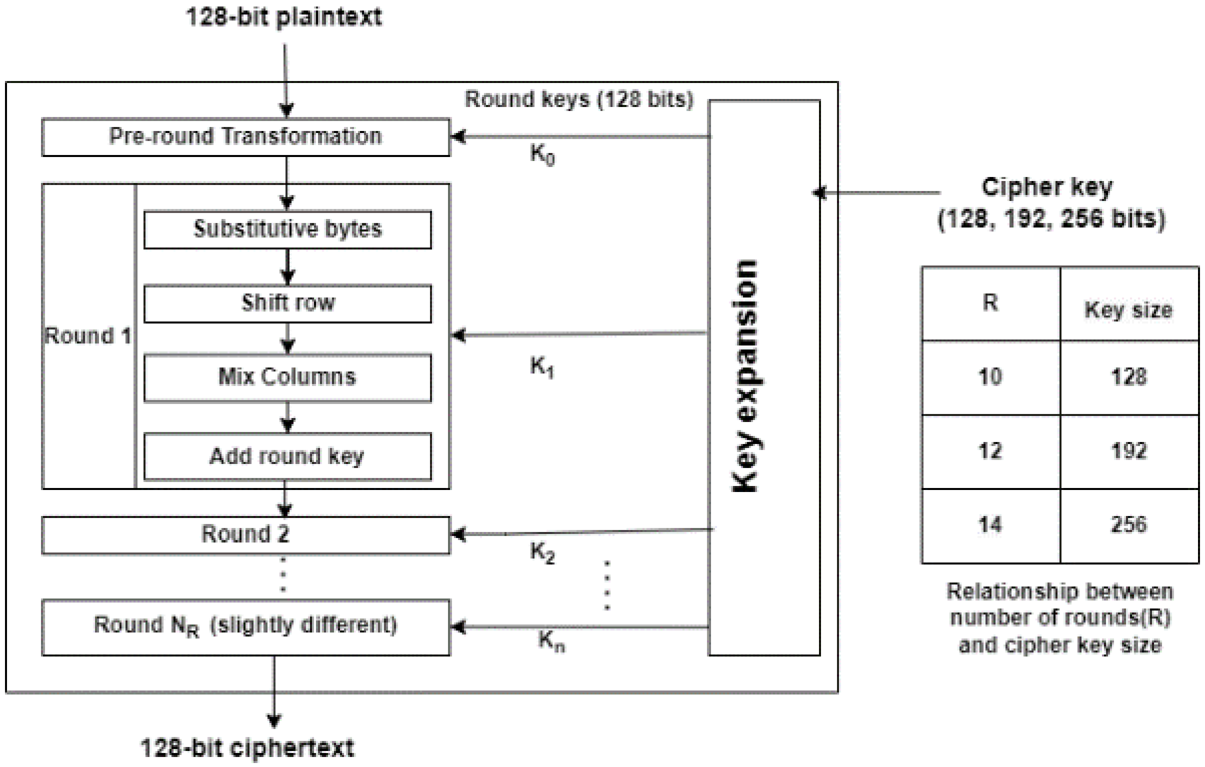
LITERATURE REVIEW

Diffie Hellman Key Exchange


Alice
Alice and Bob agree on p and g
Bob
Selects private key (a)
Select private key (b)
Public Channel
Computes her public Key g^a mod p
Computes her public Key g^b mod p
Computes K_Alice = $(g^b \bmod p)^a$ mod p
Computes K_Bob = $(g^a \bmod p)^b$ mod p
K_Alice = K_Bob
Shared secret key

LITERATURE REVIEW

Advanced Encryption Standard



R	Key size
10	128
12	192
14	256

Relationship between
number of rounds(R)
and cipher key size

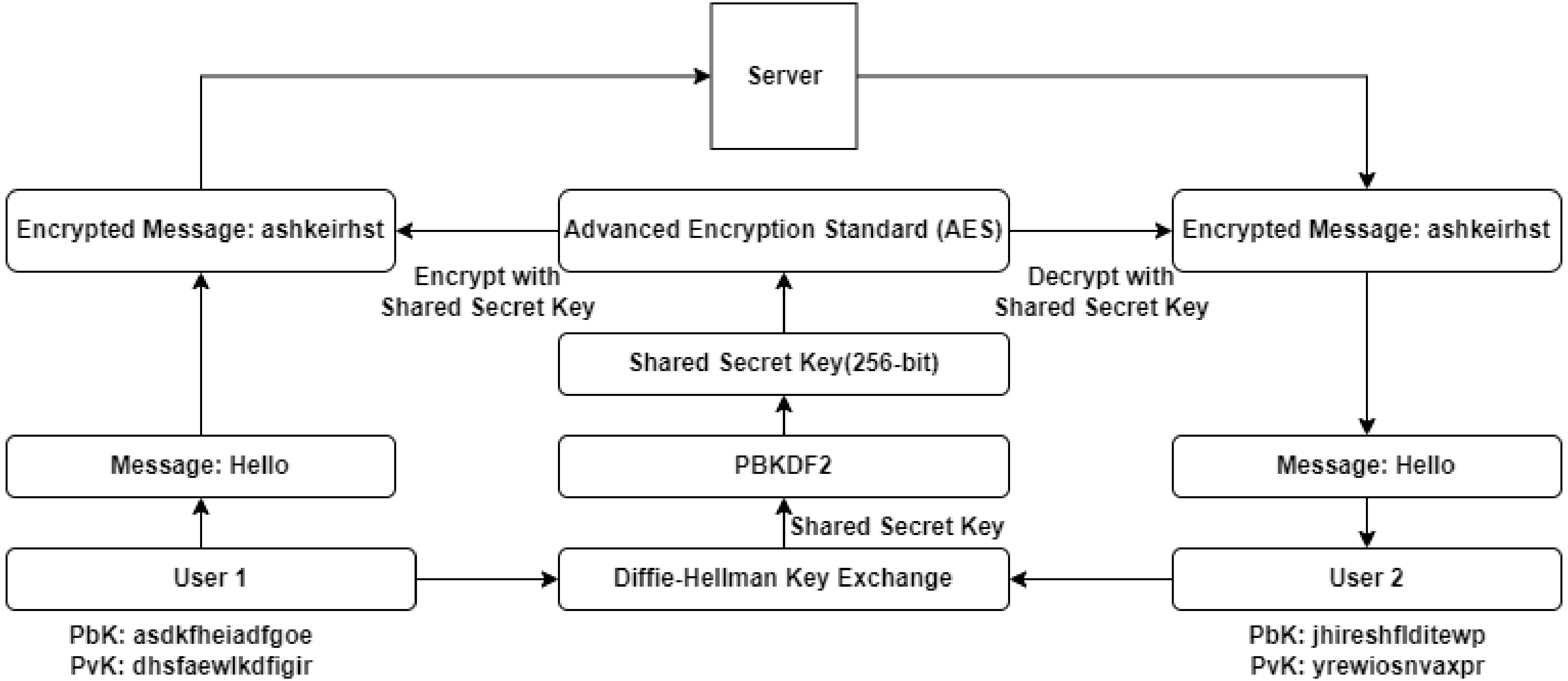
LITERATURE REVIEW

Integration with Key Derivation Function(KDF)

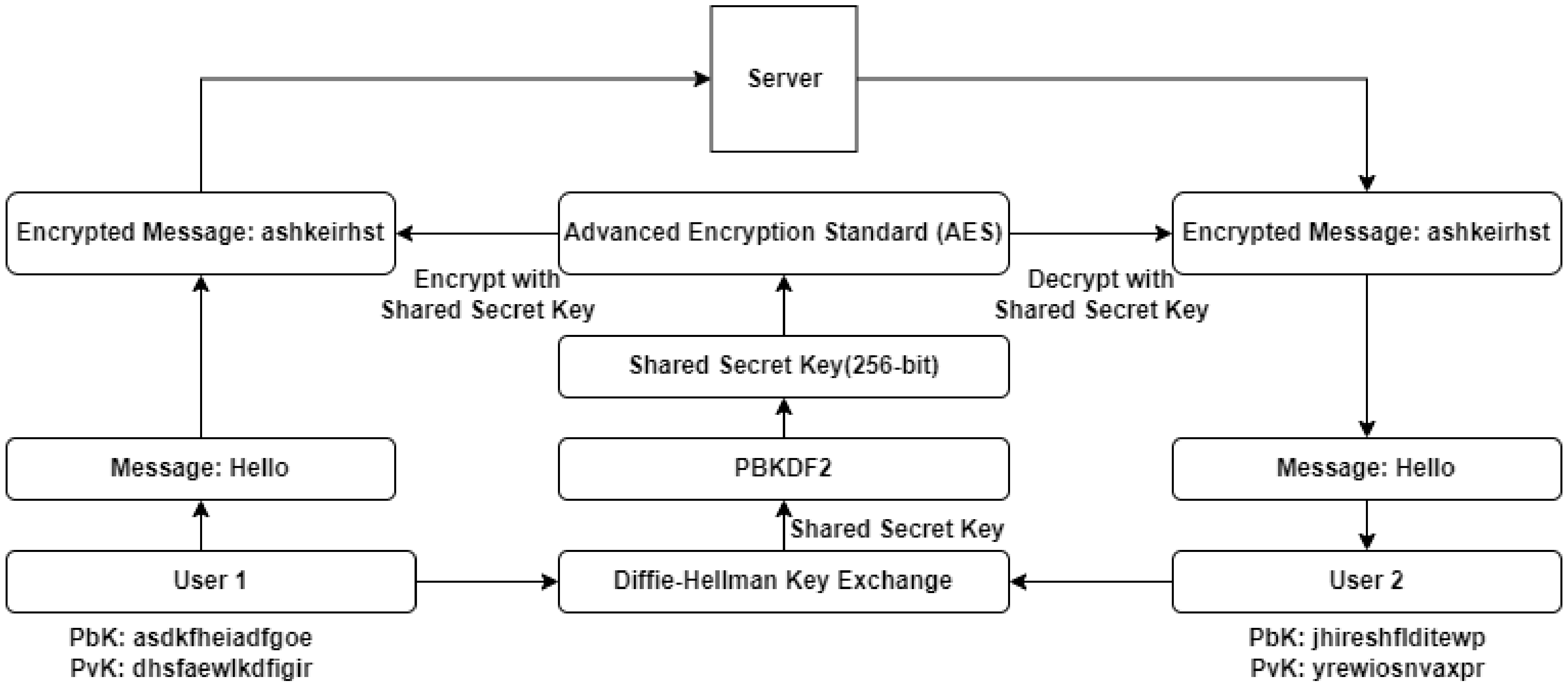
- **A Key Derivation Function is a cryptographic function that derives one or more secret keys from a secret value such as a shared secret key.**
- **A KDF typically takes as input a shared secret key and some additional parameters such as a salt and an iteration count.**
- **There are many different KDFs available such as HKDF, PBKDF2, bcrypt, scrypt, etc.**

METHODOLOGY

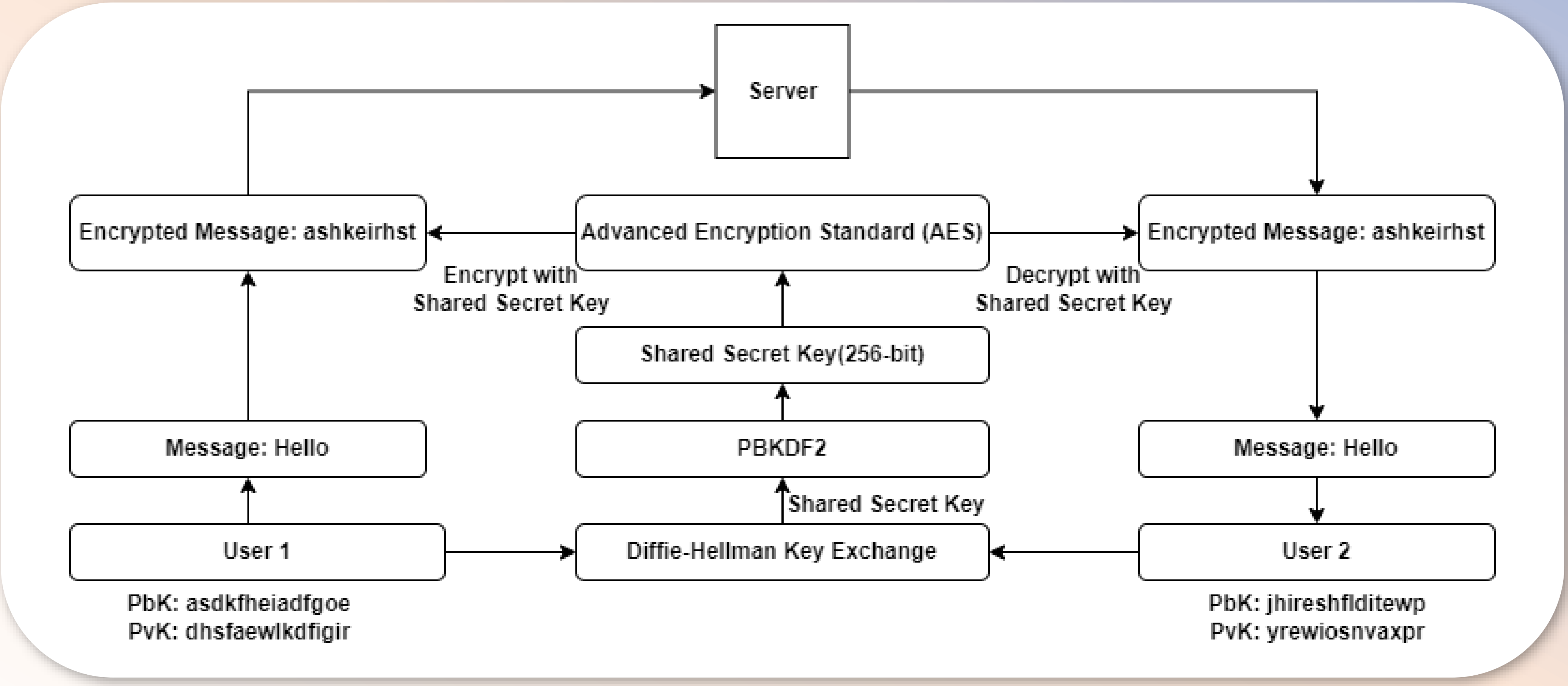
Step 1: By implementing Diffie Hellman Key Exchange Algorithm, we generate a shared secret key for the two users.



Step 2: Integration using PBKDF2



Step 3: Encryption and Decryption of messages using Advanced Encryption Standard.



RESULT

Testing for generation of shared secret key.

Sender

p: 1597

g: 11

Public key:
646

Secret key:
2211

Shared Secret key:
e74e44fb37416d1259a078e24b8491e03d5e947dc7d485b4076686287c052248

Receiver

p: 1597

g: 11

Public key:
1537

Secret key:
7647

Shared Secret key:
e74e44fb37416d1259a078e24b8491e03d5e947dc7d485b4076686287c052248

RESULT

Testing for encryption and decryption of message

Sender

Message: hello

Shared Secret key: e74e44fb37416d1259a078e24b8491e03d5e947dc7d485b4076686287c052248

Encrypted Message: U2FsdGVkX18ECSQSAJh1CYOz7M4p9cOJp+ggVX5LGIM=

Receiver

Encrypted Message: U2FsdGVkX18ECSQSAJh1CYOz7M4p9cOJp+ggVX5LGIM=

Shared Secret key: e74e44fb37416d1259a078e24b8491e03d5e947dc7d485b4076686287c052248

Decrypted Message: hello

CONCLUSION

In conclusion, implementing end-to-end encryption in a chat application is a significant security measure that provides a high level of privacy and security for users. By encrypting messages on the sender's device and decrypting them only on the recipient's device, end-to-end encryption ensures that messages are protected from interception or reading by third parties, including chat application providers or attackers.

DEMO

AUTHENTICATION

THANK YOU!