Digital Envelope for Secure Communication

CSL505: Cryptography

Abu Talha (12310020)
Vipin Kumar (42300090)

Abstract

The document presents a brief report of the project whose main objective is to build a cryptographic tool-kit for a secure communication system using a combination of asymmetric and symmetric cryptographic primitives. Typically, it is known as Digital Envelope which is widely applied in the various protocols which we are using thousands of times without even realising while surfing over the internet. We implemented an instance of Digital Envelope by invoking Advanced Encryption Standard (AES) scheme and a public-key encryption scheme RSA (Rivest-Shamir-Adleman).

Motivation

The motivation behind this project lies in addressing the critical need for secure communication in today's digital age. With the increasing importance of data privacy and security, the project aims to provide a practical implementation of widely-used encryption algorithms to ensure the confidentiality and authenticity of sensitive information.

We address a problem that asks to solve the following:

1. There are two-parties Alice and Bob. Alice wants to send a large-file (we took a file of size 100 Mb) to Bob over insecure channel.
2. How Alice can send this file to Bob efficiently while achieving the privacy and authenticity?

The problem can be addressed by using cryptographic primitives. A complete solution is to build a secure communication system using hybrid cryptographic techniques as explained in the next sections.

Problem Solving Approach

The problem is solved in the following two ways and later we compared it.

1. **Solution 1:** In this method, we simply invoke RSA encryption scheme which is owned by Bob.
   (a) Bob generate RSA-key pairs (**sk**, **pk**) by key-generation algorithm.
   (b) Alice uses Bob's public-key and encrypts the 100 Mb file.

   The major problem Alice faced is that it took her approximately 28 seconds time to encrypt her file. Further Bob's also took a lot time to decrypt it later. So the above solution is not a feasible solution in today's technological era.

2. **Solution 2:** This solution uses the RSA public-key encryption scheme along with an symmetric-key encryption scheme AES.

    (a) Alice first generates a random 256-bit key to be called $\mathsf{aes}_{\mathsf{key}}$.

    (b) Alice encrypts $\mathsf{aes}_{\mathsf{key}}$ using the RSA public-key of Bob.

    (c) Then she encrypt the her file using AES and the $\mathsf{aes}_{\mathsf{key}}$.

    (d) Now she sends the package containing encrypted file, encrypted $\mathsf{aes}_{\mathsf{key}}$ to the Bob.

    (e) After receiving the package, Bob first recovers the $\mathsf{aes}_{\mathsf{key}}$ by decrypting the encrypted $\mathsf{aes}_{\mathsf{key}}$.

    (f) Then he decrypts the encrypted file using the $\mathsf{aes}_{\mathsf{key}}$ and AES.

    The above solution provides an efficient way to solve the problem. The time taken by the fundamental operations of the invoked algorithms are depicted in the Table 1

Results, Algorithms & Environment

The following section specifies the used algorithms and their operations to solve the problem using the solution approach 2. We implemented the solution 2 in python language whose source code is submitted along with this report.

- **Algorithms:**
    - **RSA:** It contains three sub-algorithms *key-generation*, *encryption*, and *decryption*. Optionally, we use RSA Digital signature scheme in order to bring the integrity, non-repudiation and authentication in the communication.
    - **AES:** An symmetric-key algorithm which is imported from python library namely cryptography along with the Electronic-code book operation (ECB)
- **Environment:**
    - Python programming language.
    - Jupyter Notebook for code development.
    - Libraries used: `cryptography`.

Conclusion

We successfully implemented a secure communication system i.e., Digital Envelope, using RSA and AES encryption schemes. The RSA encryption and decryption processes were demonstrated in python language, and large files were securely encrypted and decrypted using AES. The project also included the signing and verification of RSA digital signatures optionally to ensure the integrity, and authentication of transmitted data. The results highlight the practicality and efficiency of the implemented algorithms in a real-world scenario.

Solutions Approaches	Operations	Time Taken (Seconds)
Solution 1	RSA-Encryption	27.78
	RSA-Decryption	288.46
	Total Time	318.22
Solution 2	AES-Key Encryption (RSA)	0.00014
	AES-Key Decryption (RSA)	0.00099
	Total time (AES-Key Exchange)	0.25
	AES-Encryption of File	0.193
	AES-Decryption of File	0.200
	Total time	0.6500

Table 1: Time taken for the operations used in Digital Envelope