International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 4 Issue 6, September-October 2020 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

RSA Cryptosystem using Python

Dr. C. Umarani[1], Aditi Badnore[2]

[1]Assistance Professor, [2]Student,
[1,2]Masters in Computer Applications, Jain (Deemed-to-be) University, Bangalore, Karnataka, India

ABSTRACT

Cryptography is a technique or method for securing communications by using codes, so that the third parties cannot use that sensitive information and only intended users can read and process it. To avoid such things encryption and decryption is the solution for the same, which will ensure confidentiality, integrity, prevent information from tampering, forgery and counterfeiting. For encryption decryption process, we require a key it is mandatory that key should not be leaked, hence for this purpose we use private and public key which is only known by the receiver and sender. This paper proposes complete implementation of RSA text encryption/decryption as well as image encryption/decryption.

How to cite this paper: Dr. C. Umarani | Aditi Badnore "RSA Cryptosystem using Python" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-4 | Issue-6, October 2020, pp.1708-1710, URL: www.ijtsrd.com/papers/ijtsrd35752.pdf

IJTSRD35752

I. INTRODUCTION

The Rivest-Shamir-Adleman (RSA) was publically described in 1977, which is used by modern computers to encrypt and decrypt the messages. It is an asymmetric cryptographic algorithm. As two different keys are used in this algorithm hence the name asymmetric. One key is shared to anyone therefore it is also called public key cryptography. Public key cryptography is largely used for authentication, non-repudiation, and key exchange. RSA is a relatively slow algorithm. Due to this limitation, it is not commonly used to directly encrypt data. RSA is used to transmit shared keys, which are then used for bulk encryption-decryption. RSA involves four steps for the complete process which are key generation, key distribution, encryption, and decryption. All steps are mentioned below.

II. WORKING OF RSA

1. **Key Generation:** Each person or a party who desires to participate in communication using encryption needs to generate a pair of keys, namely public key and private key.

A. Generating the RSA modulus (n)
B. Finding Derived Number (e)
C. Forming the public key
D. Forming the private key

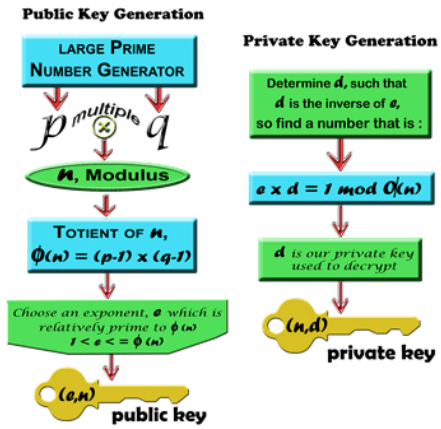


Fig1: RSA Public Key and Private Key Generation Method

2. **Key Distribution:** If a message is sent from one end to the other using RSA, one person will have the public key which is used for encryption (*n*, *e*) and other person will have the private key which will be used for decryption (*d*) and won't be distributed.

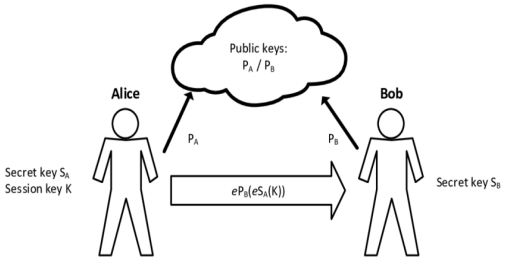


Fig2: Example of Key Distribution

3. **Encryption:** Message is encrypted using a public key (n, e). To encrypt the first plaintext P, this is modulo n. Mathematical step for encryption is – $C = P^e \bmod n$

   Ciphertext C is equal to the plaintext P multiplied by itself e times and then reduced modulo n.

4. **Decryption:** Message is decrypted using private key by the receiver. The receiver has received a ciphertext. Plaintext = $C^d \bmod n$

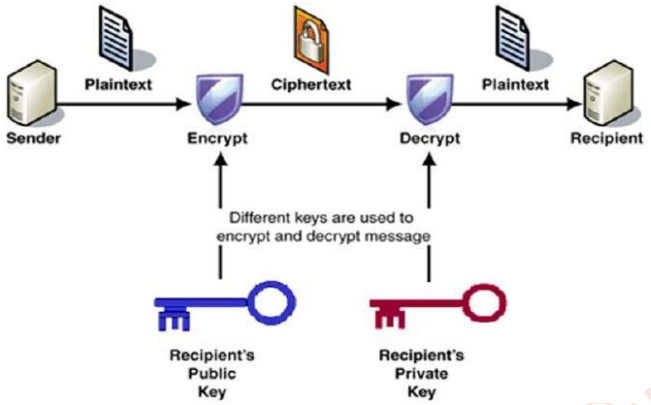
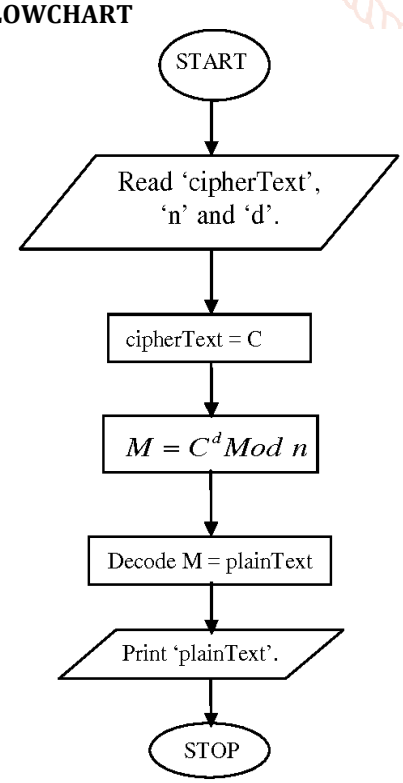


Fig3: Encryption and Decryption in RSA

III. CURRENT USAGE OF RSA

RSA is one of the most widely used algorithms amongst public key cryptography. RSA has a main advantage and is preferably known for secure communications. Earlier it was used in TLS and also in PGP encryption. At present, RSA is used in VPN, emails, web browsers. Recently, SSL also used RSA algorithm accompanied by File transfer Protocol (FTP), HTTP, Network News Transfer Protocol (NNTP) which established a key during SSL communication between the server and the client. Also, RSA can be used in employee verification. Smart card chips can be developed using this algorithm.

IV. FLOWCHART



To encrypt a plaintext M using an RSA public key we simply represent the plaintext as a number between 0 and N-1 and then compute the ciphertext C as:
$C = M^e \bmod N$.

To decrypt a ciphertext C using an RSA public key we simply compute the plaintext M as:
$M = C^d \bmod N$.

V. IMAGE ENCRYPTION AND DECRYPTION

Example can be Core banking which is a set of services provided by the group of networked bank branches. These days Internet multimedia is on highest demand.

The below pictures tell the original image which is encrypted and then again decrypted.

Fig4: Original image

Fig5: Encrypted image

Fig6: Decrypted image

VI. CONCLUSION

Though, RSA is most used algorithm these days, still it has some limitations which are getting replaced by further versions of RSA. In today's digital world the most important thing is to encrypt the image due to various types of attacks and misusing of the same. Image encryption using RSA is proved to be efficient enough and highly securable.

VII. REFERENCES

[1] https://ieeexplore.ieee.org/document/6021216/figures#figures

[2] https://www.tutorialspoint.com/cryptography/public_key_encryption.htm

[3] http://www.ijcset.net/docs/Volumes/volume5issue9/ijcset2015050902.pdf

[4] http://www.isg.rhul.ac.uk/static/msc/teaching/ic2/demo/42.htm

[5] https://en.wikipedia.org/wiki/RSA_(cryptosystem)

[6] https://www.geeksforgeeks.org/rsa-algorithm-cryptography/

[7] https://www.di-mgt.com.au/rsa_alg.html