A Hybrid Encryption Algorithm Based on AES and RSA in IOT devices

Urmil Kalaria
Student
Department of CS and Engineering
PES University
Bangalore,India
urmilkalaria@gmail.com

Pradyuman K.K
Student
Department of CS and Engineering
PES University
Bangalore,India
pradyumankannan@gmail.com

Dr. Shobana Padmanabhan
Professor
Department of CS and Engineering
PES University
Bangalore,India
shobanaau@gmail.com

***Abstract*— In the modern era of technology where lives of millions of people are dependent on devices. IoT devices play a crucial role. And security and efficiency in such devices are of prime concerns to scientists and engineers. However due to limited resources availability and security issues in such devices it makes them more vulnerable. To increase security and efficiency in transmission of data in IOT devices, a Hybrid encryption algorithm has been proposed which is based on AES and RSA algorithms. The AES algorithm is used to encrypt the plain text file with higher efficiency using block encryption whereas the RSA algorithm is used for the encryption of the AES key. Meanwhile the procedure of the entire encryption and decryption process still remains simple and efficient.**

Keywords—Hybrid Encryption Algorithm, RSA algorithm, AES algorithm

I. INTRODUCTION

There are millions of IOT devices available to us in our corporate world, which generate a tremendous amount of data in real time. For such devices there are various concerns regarding data privacy and security. The criteria to encrypt this data is hence a serious challenge because these IOT devices consist of extremely low resources than the one commonly used in Personal Computers. Recent advancement in Cryptography has created us with a separate branch to deal with such devices, called as lightweight Cryptography which is specifically designed to overcome the above problem.

AES algorithm is one such lightweight algorithm which requires lower memory and is easy to implement on both hardware and software. However the RSA algorithm also enhances security over transmission protocols to get a better blend of efficiency and security. In this research, there are three main components: Cipher text, Plain text and Keys. The plain text comprises data that is generated by IoT devices and that is required to be encrypted because of valuable metadata that are embedded in it. This plain text is going to be encrypted using AES algorithm. Hence the plain text is going to be converted into Cipher text using the block encryption method. The key used for plain text encryption is further encrypted using RSA algorithm.

II. THE ENCRYPTION ALGORITHMS IN IOT DEVICES

Cryptography is a Science of converting a message into an encrypted form which is not readable to humans and computers. There are basically two types of encryption algorithm: Symmetric algorithm (One-Key encryption) where only one key is used to encrypt and decrypt the data, and asymmetric algorithm (Two-key encryption) where Public Key is used for encryption and Private key is used for decryption.

Some of the important cryptographic algorithm available for IOT devices are Rivest-Shamir-Adleman(RSA) Algorithm, Data Encryption Standard(DES) Algorithm, Advanced Encryption Standard(AES) Algorithm, Tiny Encryption Algorithm(TEA) Algorithm, RS2 Algorithm, Blowfish Algorithm, etc. These algorithms are very good in information security, but they consume a significant amount of resources. The complexity of these algorithms makes them secure but also increases computation time and memory utilization which is limited in case of IoT devices. However the above problem has been addressed by reducing the complexity which lies in the key size used. The strength of a Symmetric Algorithm is based on its key size eg: (32bit, 64bit or 128 bit ETC). With lesser key size the encryption time reduces and also the resources used for encryption also reduces and makes fit for IoT devices.

Among the various Encryption Algorithms supported by IOT devices, we have specifically chosen Advanced Encryption Standard algorithm (symmetric) which is the most attractive among all. The various advantages of this algorithm compared with others is that it consumes less memory and is easy to implement along with the data security which seems to be evident.

However, for Asymmetric algorithm we have chosen RSA encryption algorithm which is comparatively new and more efficient and secure as compared to other asymmetric algorithms, and most importantly it is supported with the devices which have extremely low resources. Hence for our hybrid encryption algorithm we have chosen AES and RSA.

A. Comparing Various Algorithms for Hybrid Compatibility

LabView is a system engineering software used for the testing application and measuring hardware utilization for that software. With the help of LabView we had done performance testing and among various algorithms and

come up with the most effective algorithms for the hybrid algorithm.

There are various benchmarks on which the algorithms are tested and chosen from. The benchmark considered here are as follows:

2.1.1 Time Complexity Analysis

When Different algorithms are tested on files of different sizes then time taken to encrypt the data is noted and this is the benchmark for Time Complexity testing.

The below table consists of six text files of different size that are encrypted using different algorithms and corresponding time is noted for encryption.

File Size	Encryption Time in MiliSeconds				
In KiloBytes	AES (256-bit)	DES (56-bit)	3DES (128-bit)	RC2 (128-bit)	RSA (2048-bit)
Text File1 (915 kB)	2050	2133	2235	2064	4915
Text File2 (5.384 MB)	3606	3255	3094	3867	7805
Text File3 (11.804 MB)	4882	6481	7062	7847	15440
Text File4 (35.35 MB)	13371	16735	18337	21505	39107
Text File5 (59.809 MB)	25038	32411	38483	39958	65527
Text File6 (106 MB)	51249	65721	78041	83432	109237

2.1.2 Conclusion:

On performance analysis of various Algorithms we could see that the AES algorithm is best suited for a symmetric encryption algorithm since it consumes least power.

However in the case of Asymmetric algorithms we choose RSA over other algorithms because of its security and complex mathematics involved that makes it almost impossible to crack. But on comparing the time complexity we find that RSA is more expensive to use but in our case we are encrypting the key of the symmetric algorithm only, which is 128 bit key, so there is no significant change due to time complexity while using RSA Algorithm.

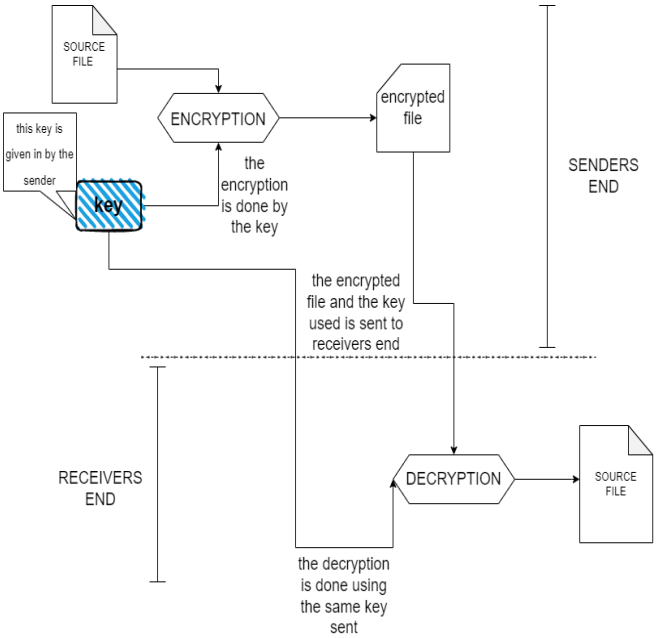
B. Working With The Symmetric Key Encryption

The encryption and decryption of the text file is done using a single key. The goodness of these algorithms is mainly based on how securely the key is exchanged between the sender and the receiver.

On the bases of encryption these algorithms are further classified into block and stream ciphers. Stream cipher encrypts the data one bit at a time continuously whereas the block cipher converts a group of bits at a given time. Block ciphers are generally faster than our stream cipher.

Different IOT devices have different computational abilities, the major requirement of efficient security is that it should be lightweight. The time taken by the protocol should be very less. Also the security algorithm must be very easy to understand and must have minimal overhead, because of this reason most of the common algorithms don't work well with our IOT networks.

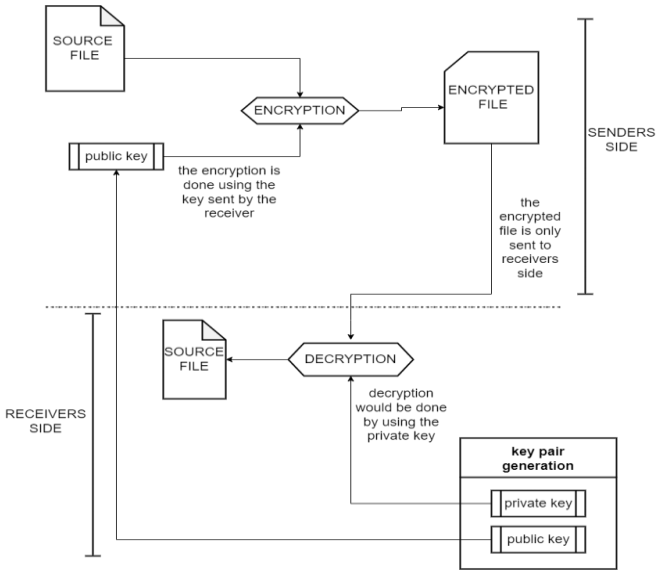
The Advanced Encryption Standard algorithm (AES) has become the most popular in our modern world.



C. Working With The Asymmetric two key encryption

Asymmetric algorithm indeed utilizes two different keys namely public and private key. The generation of these keys is done by the receiver. The public key is available to all, whereas the private key is restricted with the receiver only. The sender receives the public key and uses this key to encrypt the source text file and send the encrypted file back to the receiver. The receiver now can decrypt the encrypted source file using the private key.

Generally these algorithms are a bit complex and utilize a lot of resources, thus most IOT devices use symmetric algorithms.



2.3.1 Detailed description of RSA Algorithm:

The idea behind the working of RSA algorithm is based on the difficulty to factorize a large integer. Here there are two keys, namely Public Key and Private Key.

Public key consists of multiplication of two large prime numbers. And the private key is derived from the public key. So if anybody factorizes the public key, the private key is compromised. Thus the strength of this Algorithm lies in the key size which is basically 2048 bit long.

The plain text is represented as numbers and then some complex mod operation is applied using the public key to encrypt the data. However to decrypt the data one needs private key only, public key won't work this makes them extremely secure.

III. HYBRID ENCRYPTION ALGORITHM

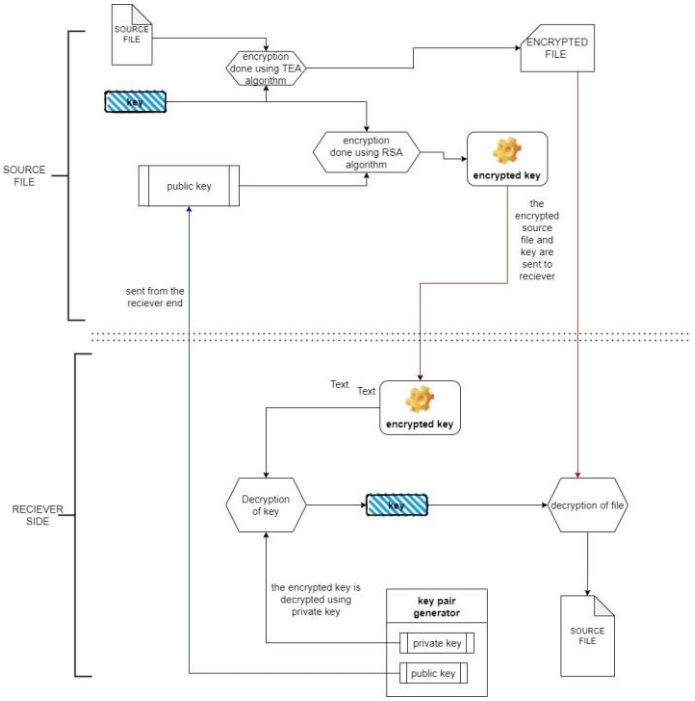
In our research we have proposed an algorithm which is a combination of both symmetric and asymmetric algorithm. Symmetric Algorithm has been used to encrypt the plain text document. The advantage of the above step is that memory utilization as well as time complexity is less as compared to using only asymmetric algorithms for encryption of the plain text.

During the encryption using AES algorithm, the sender encrypts the file using his random personal n-bit key, which is completely different from the key pair generated in the asymmetric algorithm. Now on the receiver end, the receiver will generate a key pair using the RSA algorithm.

Receiver will send his public key to the sender, even if the public key is somehow compromised also, the hacker won't be able to decrypt the message because for decryption a private key is used which is completely secured with the receiver. With the public key sent by the receiver, the key of the symmetric algorithm is encrypted using the RSA algorithm.

A. The advantages of hybrid encryption algorithm:

- Using RSA and AES algorithm for data transmission, the key transferred problem in symmetric algorithm is resolved and security is enhanced
- The algorithm implements verification using digital Signature and tags.
- The speed of encryption and decryption is almost similar with the speed of any symmetric algorithm.



Now both the encrypted file and the key will be sent to the receiver. On the other hand the receiver will use his private key to decrypt the message.

B. Safety analysis

The safety of hybrid encryption algorithm is based on speed and efficiency of encryption and decryption by AES algorithm. AES uses a n-bit key, whose decoding speed is faster.

In RSA algorithm the security is largely based on integer factorization, thus making it very difficult to crack. In conclusion as long as the security of the private key is maintained, data security can be guaranteed.

C. Performance Analysis of Hybrid Algorithm:

In the Hybrid algorithm, we encrypt the file using symmetric encryption that could take nearly 2.2 milliseconds for a file of size 26.7 kB. However, the time required for the key generation and encryption of symmetric keys using RSA is always constant because the symmetric key length is fixed

i.e 128 bit. So the time required for encryption is slightly more than a symmetric algorithm but significantly less than an asymmetric algorithm. Also the efficiency and security under the protection of dual algorithms is increased to a great extent which is in excess with both symmetric and asymmetric algorithms separately.

IV. Conclusion

The proposed Hybrid of two Symmetric and Asymmetric algorithms will make the data transmission and data security safer and more efficient. This algorithm is designed keeping in mind the constraints and limitations of IoT devices. With the minimum resources the output of this algorithm is maximized and made suitable for IoT devices.

V. Future Enhancements

The algorithm can yet be extended with other types of data like embedding the document in video or image. Also the image file can likely be converted into a binary text file and encryption can be applied to it. Our future work will focus on above mentioned problems as well as more towards statistical analysis of algorithms as compared to other hybrids.

References

[1] https://ieeexplore.ieee.org/abstract/document/5558315

[2]https://www.researchgate.net/publication/331324320_A_Secure_and_Efficient_Lightweight_Symmetric_Encryption_Scheme_for_Transfer_of_Text_Files_between_Embedded_IoT_Devices

[3]https://ieeexplore.ieee.org/abstract/document/8378034

[4]https://dl.acm.org/doi/abs/10.1145/3374664.3379531

[5]https://www.researchgate.net/publication/342853514_Honey_Encryption_based_Hybrid_Cryptographic_AlgorithmA_Fusion_Ensuring_Enhanced_Security.

[6]https://iopscience.iop.org/article/10.1088/1757-899X/745/1/012039

[7]http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.680.393&rep=rep1&type=pdf

[8] https://ieeexplore.ieee.org/document/8706017

[9]https://www.researchgate.net/publication/312129072_Enhanced_RSA_algorithm_for_data_security_in_cloud?enrichId=rgreq-825ea62ed28673277ea5d82316a4d78a-XXX&enrichSource=Y292ZXJQYWdlOzMxMjEyOTA3MjtBUzo3NDg5MzE2NDY1MDQ5NjFAMTU1NTU3MDYyNzIzNQ%3D%3D&el=1_x_3&_esc=publicationCoverPdf