International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 01 | Jan 2020 www.irjet.net p-ISSN: 2395-0072

Message Encryption using Hybrid Cryptography

Anirudh K[1], M Ramachandra Kashyap[2], K Radha[3]

[1,2]IV-CSE, CSE,GITAM UNIVERSITY, Patancheru, Rudraram, Telangana, India
[3]Asst Professor, CSE, GITAM UNIVERSITY, Patancheru, Rudraram, Telangana, India

---

Abstract - *Communication through Messages has become a major and effective part of our day to life. But, as the amount of available data grew, so did the sensitivity of the data at hand. So, it is necessary to make sure that the Data that is being transmitted via messages is usually safe from unauthorised and third party sources. That is where encryption comes into play and there exists multiple encryption algorithms to serve the purpose. But, using only one algorithm for this, is generally not secure enough. So, by using two effectives encryption standards (AES and Triple DES), we aim to provide a lighter, easy to understand and effective approach to Message encryption. Since, we are using more than one techniques of encryption, its called hybrid cryptography. Since, the algorithms used are symmetric key encryptions, the users are required to transfer a key within them prior. Then , the message is split into two equal halves and each half is encrypted using any one algorithm and the previously acquired key. These encrypted parts are transmitted as a single message and then decrypted at users side to get the original message back.*

Key Words: Cryptography, Encryption, AES (advanced encryption standard), DES (Data encryption standard), Key (secure variable)

1. INTRODUCTION

Cryptography technique translates original message into a sequence of unreadable symbols and characters. The Cryptography techniques available in the market are broadly divided into two types, they are symmetric key cryptography and public key cryptography. These techniques uses keys for encrypting the data into unreadable form, so that only authorized persons that have the pre agreed key can actually decrypt and view the message that is being transmitted. The techniques that we are applying in our system both fall under the symmetric key cryptographic techniques. This involves transferring a secure, private key between both the parties prior to establishing the communication path via any secure source like Mail.

So, since no two algorithms are similar in nature, there are different requirements for different algorithms. For example, DES can only work with a 56 bit key and AES on the other hand can work with 128, 192 or 256 bit keys. The plain text or the initial message to be encrypted in DES should not exceed 64 bit block size where as AES can work with larger block sizes as well. So, when we are attempting to try and incorporate these techniques as a single algorithm, we would need to keep track of these anomalies and make sure that no ambiguities arise in the implementation.

To handle varying input sizes and key sizes, the approach of the algorithm has been modeled in the given way, initially, the key is expected from the user and if the key does not meet the minimum size, usually multiples of 16, we use the concept of padding to increase its size. That is, we append trailing zeros to the content and then use it as the key. In case of the message, the text is broken down into two equal halves and each of the halves are encrypted using any one of the algorithm. Once again, if the plain text doesn't meet the minimum size requirements of the algorithm require, we perform padding. Once, both parts of the message are encrypted , it is converted into a arbitrary sequence of symbols then the encrypted parts are combined into a single message and transmitted to the receiver. Now, the user performs decryption with the prior obtained key via secure sources using both AES and DES and ultimately gets the plain text or initial message that was transmitted.

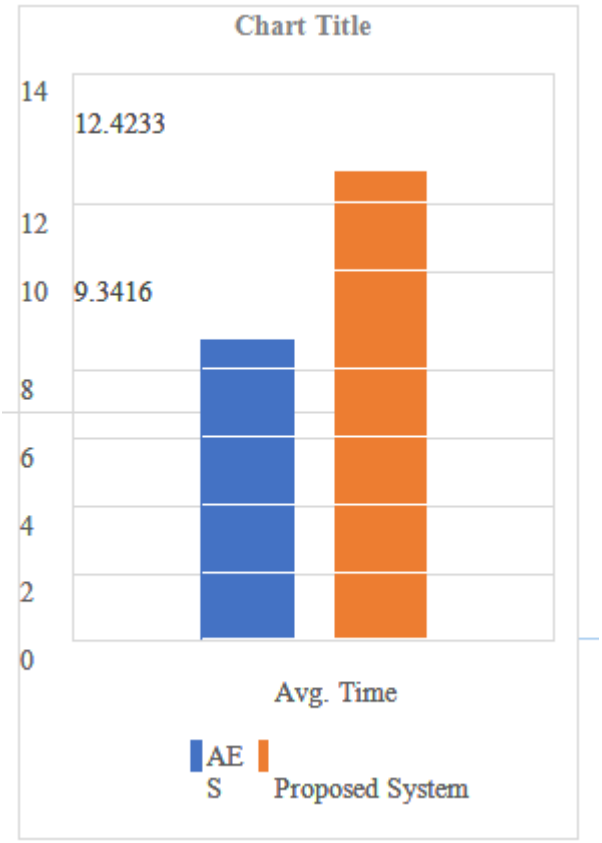
One main advantage of this approach is that even if the network is compromised and an attacker manages to get access to the encrypted part of message, it appears gibberish and without the private key, the attacker would be clueless to what the actual message might be. The reason AES and DES were chosen is because, symmetric key encryptions are easy to understand and implement and other encryption standards like mono or poly alphabetic ciphers were broken by Brute Force attacks that were possible with the advent of modern computing technologies. AES is also currently the go to Encryption technique for applications like WhatsApp, but to prioritize simplicity, we have gone with DES along with AES to make it a lighter and effective approach.

2. BACKGROUND STUDY

The basic ideology of this project came from background study of the Hybrid Cryptography algorithm paper written by Punam V. Maitri and Aruna Verma. Where, they had used AES, Blowfish and RC6 to from the hybrid cryptography and Image steganography technique to securely transfer the key from one user to the other. So, we in an effort to simplify and make the above said approach lighter have gone for a simpler way of transferring key.

3. RESULT ANALYSIS

The time taken for execution for proposed system and AES:



Even though here we can observe that AES alone performs better, AES is known to use a large number of rounds of ciphers to encrypt the content. So, for our input size it might perform better, but as the size of the input increases, AES can get considerably slow. Moreover, AES works fine as a standalone algorithm for large applications, but for simpler purposes like ours, using it on entire message would only lead to it using up the resources as it needs more number of rounds than DES.

Another question that might arise is why not use any other algorithm into the mix like RC6, the reason being that of the used algorithms, AES, 3DES and RC6, RC6 has the highest throughput thus delaying the process which can especially be frustrating for larger input sizes. Moreover, since Python was used as the language of choice, additional changes can be made by the user to meet the required specifications.

4. FUTURE SCOPE

This idea and approach that we have implemented do not end here. There can be a future scope for development and we can extend this particular project into multiple platforms. Particularly, we can incorporate additional algorithms into the mix to make it more appealing.

One other source of improvement is expanding the idea into cloud based systems so that applications and web sites can make use of the algorithm at hand to secure their way of communication between their users and themselves. Especially, since the data that is stored on clouds or any other remote storages is sensitive, an added level of security never goes in vain.

5. APPENDIX

1. SHOWING THE ACTUAL APPROACH USED IN PYTHON:

```
key = input("enter key ")

l=len(key)

if l<16:

  r=16

keyA = key.ljust(r, '0')

keyA=keyA.encode()

string = input("enter plain text ")

l=len(string)

aes_str, des_str = string[:l//2], string[l//2:]

final_encrypt= aes_en + des_en
```

6. SHOWING THE APPRAOCH TO CALCULATE THE TIME

Using a Python time library to measure the time taken for executing

```
Import time

# starting time start

=time.time()  end=

time.time()

print (end-start)
```

7. REFERENCES

[1] Punam V Maitri, Aruna Verma, Secure File Storage in Cloud Computing Using Hybrid Cryprographic algorithm, IEEE, WiSPNET-2016.

[2] V S Mahalle, A.K. Shahade, "Enhancing the Data Security in Cloud by Implementing Hybrid (RSA & AES) Encryption Algorithm, IEEE, INPAC, pp 146-149,Oct. 2014.

[3] S. Hesham and Klaus Hofmann, "High Throughput Architecture for the Advanced Encryption Standard Algorithm" IEEE, International Symposium on Design and Diagnostics of Electronic Circuits & Systems, pages 167170,April 2014.