ISSN XXXX XXXX © 2018 IJESC

Research Article Volume 8 Issue No.3

Performance Analysis of Data Encryption Algorithms using AES BLOWFISH and SNAP

Bhawana Dakhare[1], Nilesh N. Shinde[2], Swanand S. Salvi[3], Aniket H. Kadam[4], Pooja G. Wagh[5]
Assistant Professor[1], BE Student[2, 3, 4, 5]
Department of Information Technology
Bharati Vidyapeeth College of Engineering, Belpada, Navi Mumbai, India

Abstract:
Providing security for critical data and availability of that data at rest, in motion and in use is important for a cloud provider. Several alternatives exist for storage services, while information confidentiality solutions for the database as a service paradigm are still immature. We propose a novel architecture that provides cloud services such as data confidentiality, integrity and availability on encrypted information. Our proposed architecture shows the theoretical analyses and extensive experimental results based on a product implementation for different numbers of users and network latencies. For encryption a constant encryption calculation is utilized. So a client who has the key for decoding can perform unscrambling of information and access that information. Client confirmation is likewise improved the situation the framework. We have likewise contrasted this strategy and the technique beforehand executed i.e. encryption and decoding utilizing AES, BLOWFISH and SNAP algorithms. We have actualized strategy in which OAuth does the validation and give one kind of approval token to every client which is utilized as a part of encryption method that give information protection to all clients. The Real Time encryption calculations utilized for securing information utilizes the key that is created by utilizing approval token.

Keywords: OAuth 2.0, RSA key generation, AES, BLOWFISH, SNAP algorithm, encryption/decryption, cloud, data security

I. INTRODUCTION

Cryptography protects the information by transforming it into an unreadable format. The main goal of cryptography is to keep the data secure from unauthorized access. Data in the readable format is called plaintext. Encrypting plaintext results in unreadable gibberish called ciphertext.The number of security goals is provided by cryptography to ensure the privacy of data. Due to the great security advantages of cryptography it is widely used today. The required basic definitions and concepts in Cryptography are reviewed here.

- **Plaintext:** An original message is called Plain text or clear text.
- **Encryption:** Process of modifying a message to hide its substance.
- **Ciphertext:** An encrypted message is Ciphertext.
- **Decryption:** Process of turning Ciphertext back into Plaintext is called Decryption.

Cryptanalysis: Cryptanalysis is the science of recovering the plaintext of a message without access to the key.

- **ALGORITHMS USED:**
  - ➢ AES
  - ➢ Blowfish
  - ➢ SNAP
- **Advanced Encryption Standard (AES)**

The AES has the latest generation of block ciphers It helps to increase the block size - from the old standard of 64-bits up to 128-bits and keys from 128 to 256-bits. In part this has been driven by the public demonstrations of exhaustive key searches of DES and RC-5 (at 64-bits).

It has the following attributes:
1. 128-bit block size of data
2. 128, 192, or 256 bit key size
3. AES uses an iterative model rather than a Feistel cipher (like IDEA)
4. AES treats data as groups of 4 bytes. It has 9, 11 or 13 rounds, where each round consists of:

- a byte substitution step (1 S-box used on every byte)
- a shift rows step (shuffle the bytes between groups)
- a mix columns step (matrix multiplication of groups with each other)
- an add round key step

5. All operations can be combined into XOR and table lookups - hence implementation can be very fast and efficient.

- **Blowfish**

Most of the encryption algorithms today are unavailable to the public - many of them are protected by patents (e.g. Khufu, REDOC II, and IDEA), or being kept secret by the governments (e.g. Skipjack and Capstone are protected by the U.S. government). Bruce Schneier is one of the world's leading cryptologists who designed the Blowfish algorithm [5] and made it available in the public domain. Blowfish is a variable length key, 64-bit block cipher. The Blowfish algorithm was first introduce in 1993, and has not been cracked yet. It is also noteworthy to point out that this algorithm can be optimized in hardware applications, although it, like most other ciphers, is often used in software applications. Though Blowfish removes the disadvantage of AES by decreasing the execution time, security wise Blowfish is not that strong. The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. There are four 32-bit S-boxes with 256 entries. The outputs are added with modulo 2^{32} and XORed to produce the final 32-bit output

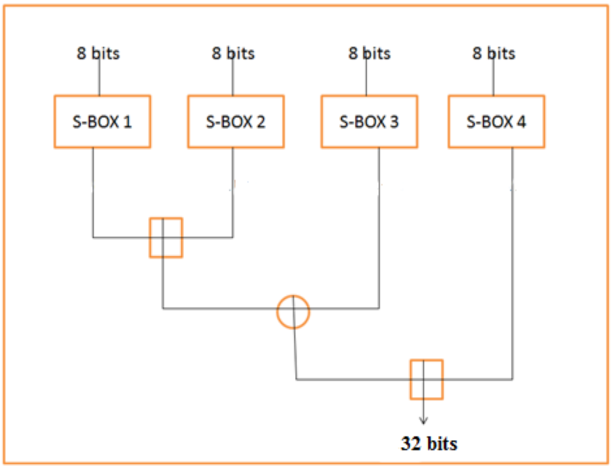

Figure.1. Existing F-Function in Blowfish Algorithm

- **SNAP:**

The original Blowfish Algorithm is improved by changing its F function to different cases. After analyzing those changes to F function of Blowfish algorithm, we will see that most of the changes makes original Blowfish Algorithm most secure. There are four cases of F function with two ADD and one XOR or with two XOR and one ADD operation. Out of this we took one case and this modification in the Blowfish is named as **SNAP**. This modification supports the parallel evaluation of two addition operations (S1, a + S2, b mod 232) and (S3, c + S4, d mod 232) by using threads. The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. There are four 32-bit S-boxes with 256 entries. The parallel evaluation reduces the time from two additions to time required for one addition. As the algorithm uses 16 iterations, this time is saved 16 times for every encryption/decryption. This is a considerable improvement. Also, as the security of Blowfish lies in the fact that it uses variable key, this modification does not make the algorithm vulnerable in any way so that cryptanalysis becomes easy. Also it does not violate any of the security issues. The modifications leads to the simultaneously executions of 2 Addition operations.

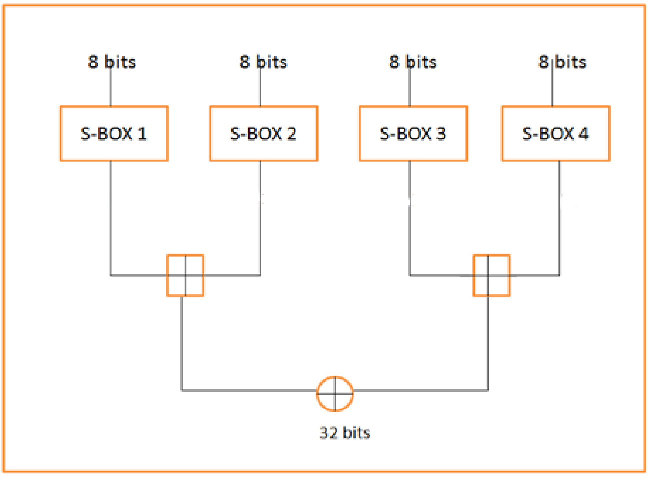

Figure.2. Modified F-Function of Snap Algorithm

In the case of original F-function which execute in sequence order and it requires 32 -addition operation and 16 XOR operations. But in the case of our modified F-function it requires the same 48 gate operations (32 addition and 16 XOR operations) but time taken to execute this 48 operations will be reduced because of multithreading. We executed 32 addition operations in parallel order using threads and hence time taken to complete 16 gate operations will be equal to the time taken 32 add operations, since we are running it in parallel environment.

II. PROPOSED SYSTEM

To overcome the disadvantages of existing system we will be proposing a secure scheme for data storage and user authentication. In this project, security for data will be implemented using encryption of file which is to be stored in cloud. For encryption three real-time encryption algorithms are used i.e AES RIJNDAEL, BLOWFISH and SNAP. So a user who has the key for decryption can perform decryption of data and access that data. User authentication is also done for the system using OAuth protocol. We have also compared and analyzed the three symmetric encryption algorithms i.e. encryption and decryption using AES, Blowfish and SNAP. Encrypting using AES results into growing of file size to double of original file and hence file upload time also increases. Blowfish used in this project removes this drawback. RSA algorithm for providing more security to data as well as our data hiding method. In this RSA will generate a secret key. We will be implementing method in which OAuth does the authentication and provide unique authorization token for each user which is used in encryption technique that provides data privacy for all users. The Real Time encryption algorithms used for securing data uses the key that is generated by using authorization token. To authenticate user we have used OAuth 2.0, which returns unique token for each user who attempts successful login. The token returned by OAuth server utilized as a part of encryption strategy so it gives information privacy and integrity to the user data. The files are encrypted before load and decrypted when job execution is in progress. The Real Time Encryption Algorithm utilizes the OAuth token as key and Encrypt data (uploaded by user) by XoR- ing with the key.

III. SYSTEM ARCHITECTURE

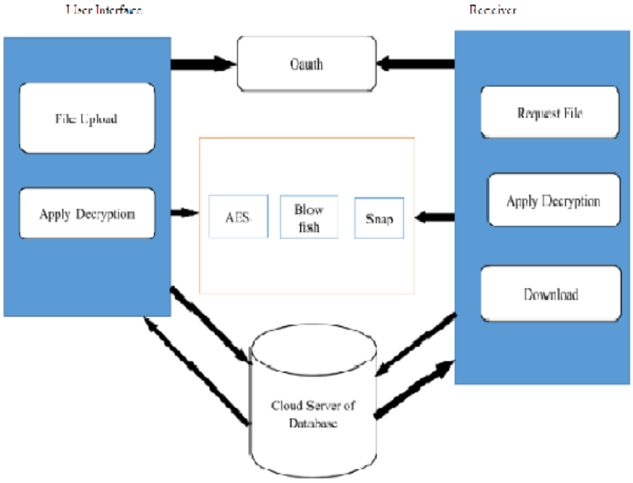

Figure.3. Architecture Diagram

CLIENT REGISTER AND LOGIN

In this module, Client wants to login. So First he registered to cloud with his own details, such as username, password, mobile no and address. Then he login with his username and password. An access token is generated. Client chooses whether to give access to your application. Then the Client is redirected to your application by OAuth Server. We can Exchange authorization code for refresh and access tokens.

The response is then processed and the generated token is stored. Then the Server verifies credentials and grant access to your application. Client is redirected to your application by OAuth Server. Validation of the client's token is done and Token validation response is processed. Then he got his client form. This client form contains two contents. One is uploading another one is downloading.

UPLOAD FILE, ENCRYPT FILE USING AES, BLOWFISH AND SNAP

A secure type is composed of three fields: data type, encryption type, and field confidentiality. The combination of the encryption type and of the field confidentiality parameters defines the encryption policy of the associated column. The data type represents the type of the plaintext data (e.g., int, varchar). The encryption type identifies the encryption algorithm that is used to cipher all the data of a column. The data is encrypted using 3 algorithms differently i.e AES , BLOWFISH and SNAP. Then generate the Key for encryption, next encrypt the file and upload to the cloud.

GENERATE KEY USING RSA

The Application will generate a secret key using RSA Algorithm. RSA algorithm for providing more security to data as well as our data hiding method.

ANALYSIS OF ENCRYPTION

Analysis of encryption algorithms is done with respect to time consumed and increase in bit values. Thus results are formed to judge the encryption algorithms. Modified blowfish shows better results than AES.

DOWNLOAD FILE AND DECRYPT

In this module the client wants to download file. So he entered the filename. Then he gives the download request to Cloud. The request is send to the file owner where the file owner decides if request should be accepted or not. If the request is accepted then the access key is send to the requester. Finally he get the Ciphertext then he decrypt and get the original file using the access key.

IV. CONCLUSION

In this paper we have compared three data encryption algorithms based on performance, file size and time required to upload the file on cloud.SNAP algorithm has helped to reduce the time required to upload the files and provide more strength in terms of security as compared to AES .

V. REFERENCES

[1]. Cornwell Jason W, "Blowfish Survey",Department of Computer science, Columbus State university, Columbus, GA, 2010.

[2]. Tamimi A. Al., "Performance Analysis of DataEncryption Algorithms", Oct 2008.

[3]. Nadeem Aamer, "Performance Comparison of Data Encryption Algorithms", Oct 2008.

[4]. A Survey Paper on Social Sign-On Protocol OAuth 2.0 (Journal of Engineering, Computers andamp;Applied Sciences (JECandamp;AS) Volume 2, No.6, June 2013).

[5]. DhawanPriya, "Performance Comparison: Security Design Choices", Microsoft Developer Network October 2002.

[6]. AES: http://searchsecurity.techtarget.com/ definition/ Advanced- Encryption-Standard

[7]. Blowfish : http://searchsecurity.techtarget. com/definition/ Blowfish

[8]. OAuth : https://oauth.net/2/