A Novel Model to Overcome Drawbacks of Present Cloud Storage Models using AES 256 CBC Encryption

Steve Boby George
Albertian Institute of Science and Technology
Cochin University P. O., Kochi, Kerala, India

Sanjay Jaimy
Albertian Institute of Science and Technology
Cochin University P. O., Kochi, Kerala, India

Sebin Jose
Dept.of Computer Science
Albertian Institute of Science and Technology
Cochin University P. O., Kochi, Kerala, India

Edwin Daji
Albertian Institute of Science and Technology
Cochin University P. O., Kochi, Kerala, India

Agnel Antony
Albertian Institute of Science and Technology
Cochin University P. O., Kochi, Kerala, India

ABSTRACT
Personal data that could lead to identity theft is very common in today's world. Sensitive information like government documents, credit card information and bank account details are now stored in cloud storage services. In today's age, encrypting data has proven to be one of the most difficult operations. The purpose of this study is to implement a simple cloud storage model, which can help a user to encrypt files using AES 256 CBC and upload them to the cloud server.

Keywords
Encryption, cloud storage, brute force attacks, AES 256 CBC

1. INTRODUCTION
A cloud computing paradigm that stores data on the web through a cloud computing provider who manages and administers data storage as a service is known as cloud storage. It's on-demand, with just-in-time capacity and rates, and it eliminates the need to purchase and manage data storage equipment. This gives the user agility, global scale and consistency with "anytime, anywhere" data access. There are numerous users who use cloud storage services to upload data which maybe confidential like unique identification documents, driving license, medical reports and others. Present approaches are not sufficient to ensure data security for end users. Usually the data uploaded is not encrypted in any form therefore there lies a vulnerability of the data getting leaked if the service provider's server gets hacked. This can lead to identity theft, bank fraud and other illegal activities in this regard. In order to alleviate this concern, some form of encryption is required to safeguard the information that has been uploaded. This study focuses on a model that encrypts the data at the client side then further processes it and uploads it to the cloud server. The advantage of client-side encryption is that even if the cloud server gets compromised the uploaded data will remain secure since its encrypted. This paper discusses about a new approach to cloud storage and also provides a brief description about the AES encryption algorithm.

2. RELATED WORK
There are many cloud computing models suggested in the past. They are discussed below.

In a study based on Secure Cloud Storage Using AES Encryption by M.P Babitha and Babu K. R. R. [1] suggested the proposed system built on a prototype of an online file processing application. The application was hosted on a cloud infrastructure provider. Anybody can access the application from anywhere at any time over internet. Graphical user interface (GUI) was created with HTML which shows the overall system architecture and activities in proposed model. The user can upload the confidential file through file upload module. Before uploading the file gets encrypted. A secret fileID is used for future accessing and sharing. They concluded that when performance of proposed approach was analyzed based on delay. They observed that there is drastic increase in delay with increase in file size. In another study R Ashalatha, M Vaidehi [2] proposed cloud computing holds the potential to eliminate the need for setting up of high cost computing infrastructure for the IT-based solutions and services that the industry uses. Since these data centers may lie in any corner of the world beyond the reach and control of users, there are multifarious security and privacy challenges that need to be understood and taken care of. The performance evaluation of AES cryptography, which is utilized for data security, was discussed by Lee Bih-Hwang, Kusuma Dewi Ervin et al. [3] Furthermore, data encryption delay calculations revealed that bigger data sizes result in longer data encryption delay times. Reema, Princy, and Kumari S. et al. [4] proposed encryption using AES and DES but the model couldn't withstand attacks like brute force. Security in the cloud necessitates a systematic approach based on trust and entrusting protection to a reliable third party[5], which is a gargantuan goal to achieve, according to Zissis D and Lekkas D.

3. THE SECURITY ALGORITHM

3.1 Brief Introduction
The Advanced Encryption Standard, sometimes known as Rijndael, is a symmetric encryption technique that employs the Advanced Encryption Standard. The data is encrypted and decrypted using the same key. Block lengths of 128, 192, and 256 bits are supported by AES. The algorithm was created by two Belgian cryptographers, Vincent Rijmen and Joan Daemen AES encryption is extremely software and hardware friendly due to the following characteristics: All known attacks are ineffective against it,simplicity of design, as well as speed and interoperability of source code on multiple computing systems [6]. Symmetric encryption is also referred

to as private-key cryptography since the key used to encrypt and decrypt the message must remain secret so that anyone with access to it may decrypt the contents. The transmitter encrypts the data with one key, sends it (the ciphertext), and the receiver decrypts it with the same key.

3.2 Steps involved in AES

3.2.1 Byte Substitution

A simple byte-by-byte substitution uses a table of 16x16 bytes, called the S-box, holds all 256 8-bit values in their permuted form. Each byte of state is replaced by a byte that is indexed by the row and the column. For example, byte {96} is replaced by byte in row 9 column 6 which has value {90}. The S-box is made to resist all previously discovered attacks.

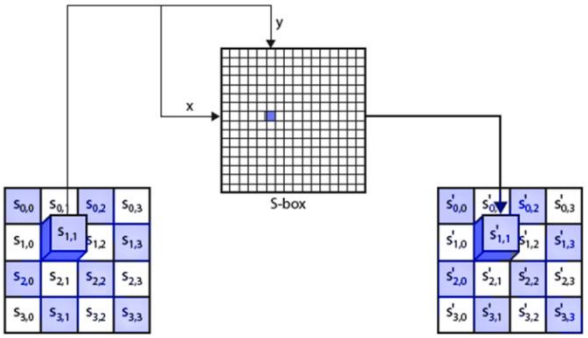

Fig 1: Byte Substitution

3.2.2 Shift Rows

Each row undergoes a circular byte shift. The first row is left unchanged, a 1-byte circular shift to the left is performed on the second row, a 2-byte circular shift to the left is performed on the third row, and a 3-byte circular shift to the left is performed on the fourth row.

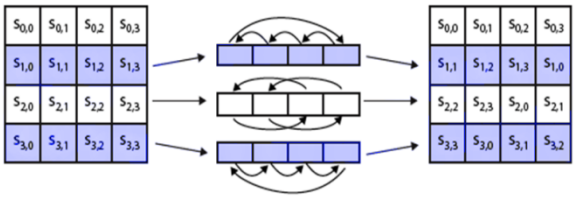

Fig 2: Shift Rows

3.2.3 Mix Columns

The forward mix column transformation, called MixColumns, operates on each column individually. [9] Each byte of a column is mapped into a new value that is a function of all four bytes in that column. Each element in the product matrix is the sum of products of elements of one row and one column. In this case, the individual additions and multiplications are performed in GF (28). MixColumns is applied in all the rounds except the last one because this doesn't make any difference in the output.

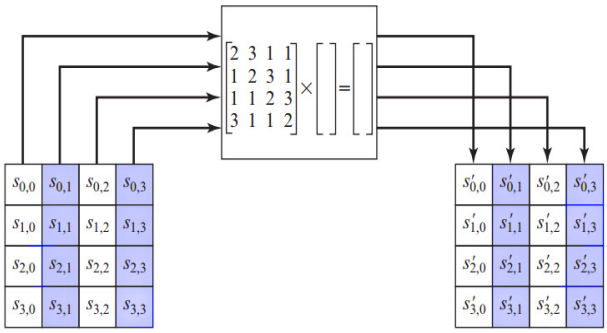

Fig 3: Mix column transformation

3.2.4 Add Round Key

The 128 bits of State are bitwise XORed with the 256 bits of the round key. The first two round keys are 256 bits taken directly from the AES key. The operation is interpreted as a column-wise operation involving four bytes from the state column and one word from the round key. The next round key is based on a simple function of the previous two round keys.

Fig 4: The forward add round key transformation

3.2.5 AES Decryption

AES decryption is not equivalent to AES encryption.Applications that require both encryption and decryption must be implemented with two separate software or firmware modules, which is inconvenient. A decrypting algorithm has the same sequence as the encryption algorithm, but a different structure (with transformations replaced by their inverses). To achieve this balance, the key schedule must be adjusted. An equivalent inverse cypher can be made by following the same steps as encryption, swapping the mix columns and adding round keys steps requires applying the inverse mix columns step to the round keys first, this complicates the decryption key schedule slightly, but allows for the use of the same hardware or software for the data encrypt and decrypt computations.

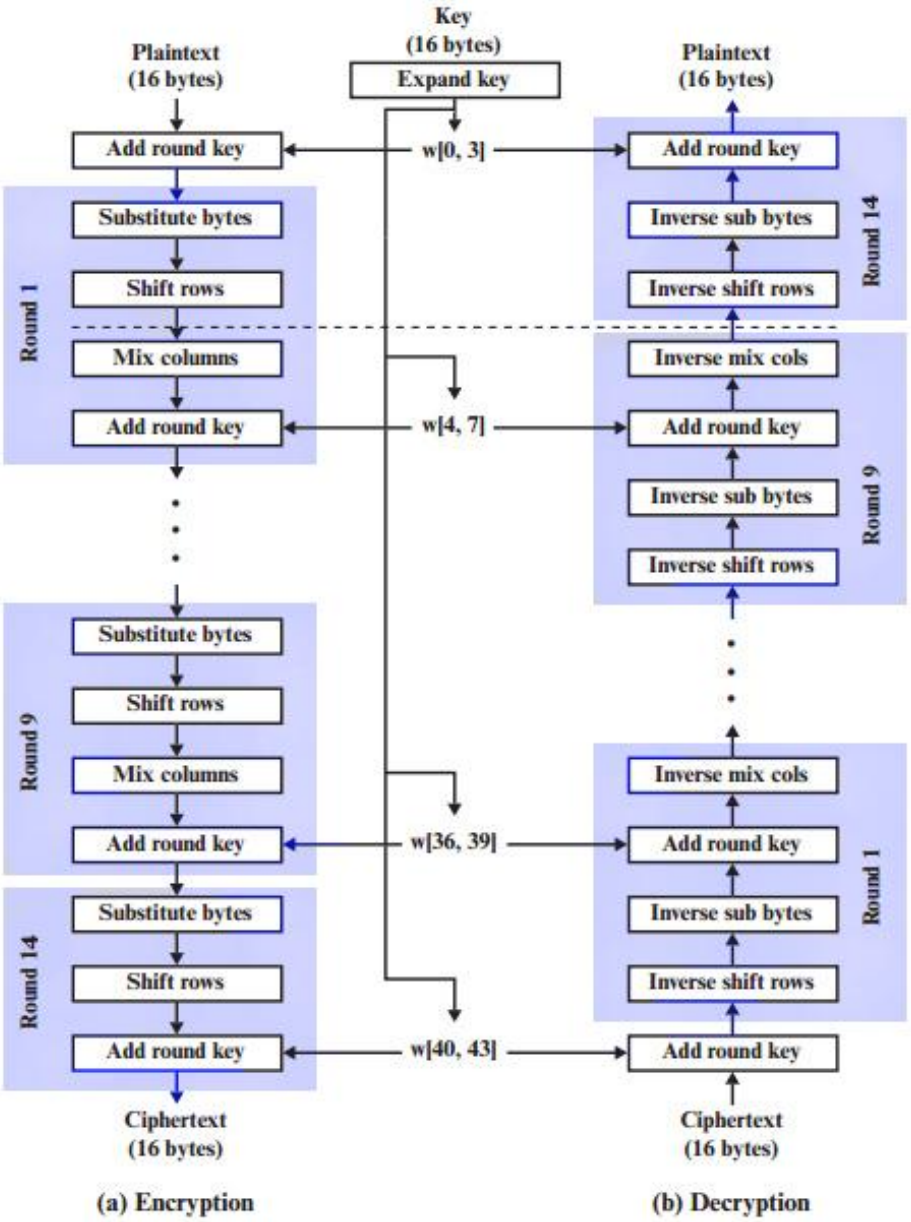

Fig 5: AES Encryption and Decryption Block Diagram

3.3 Modes of Operation for AES

Typically, AES operates under five modes [7], Output FeedBack (OFB), Electronic CodeBook (ECB), Cipher FeedBack (CFB), Cipher Block Chaining (CBC) and Counter (CTR). CBC mode is preferred because it hides away block patterns in the plaintext. The initialization vector for this mode must have the same size as the block size. Using a randomly generated initialization vector precludes identical ciphertext from being created from packets with identical data spanning the first block of the cypher algorithm's block size. Before being encrypted, XOR operation is carried out with the initialization vector and the first plaintext block. The same operation is again performed with the previous ciphertext block and the current plaintext for successive blocks. [8] Since the key length here is 256 bits it supports the largest bit size.

3.4 Benefits of AES over other encryption methods

Table 1. Key sizes and corresponding possible combinations to crack by brute force attack

Key Size	Possible Combinations
1 bit	2
2 bits	4
4 bits	16
8 bits	256
16 bits	65536
32 bits	4.2×10^{9}
56 bits	7.2×10^{16}
64 bits	1.8×10^{19}

128 bits	3.4×10^{38}
192 bits	6.2×10^{57}
256 bits	1.1×10^{77}

From Table I an observation can, be made that based on current processing power,that AES 256-bits CBC is almost unbreakable by brute force, making it the strongest encryption standard. [8]

4. SYSTEM MODEL

Before discussing about the proposed model in detail, a precise problem statement is required.

4.1 Problem Statement

Users are concerned about their security and privacy of data uploaded into the cloud. As all the cloud services are available at the remote locations, users can't have the complete control over their data. It is always their basic right to protect their data from unauthorized access. In essence, a user will upload the data into cloud using encryption technology where plaintext is changed into ciphertext. To view the unintelligible ciphertext, it needs to be decrypted using an instance of encryption algorithm called "Secret Key". This secret key is shared with the users who would like to access the data.

4.2 System Requirements

As shown in Table 2, the proposed model's system requirements include the following: Any user, whether a novice or an experienced one, can take advantage of the extremely low requirements.

Table 2. System Requirements

Hardware	Processor: Intel Pentium or above Ram: 1GB or above Storage: 100 MB of Free space Network: Active Internet Connection
Software	Operating system: Windows 7 and above. IDE: Python 3.7 and above.

4.3 Architecture of the proposed model

The client uses the GUI application to login or register to the database servers. After successful authentication the login page of the application is closed and the main page is shown where the client can select the required file for uploading to the cloud storage. After selection of the file is completed, there is a prompt for entering the encryption key. Once the key is entered, the encryption process starts locally in the client's machine. After successful encryption, the file is uploaded to the cloud storage. If the client wants to download a file from the cloud storage then the corresponding file is selected and downloaded. Once downloaded a prompt is displayed to enter the key for decryption. Therefore, after successful decryption the file can be accessed by the client. There is also a provision to delete files from the cloud storage.

Fig 6: Model Architecture

5. IMPLEMENTATION

The proposed model was implemented using the following technologies Microsoft .NET Framework, Python3 and Google Firebase. Microsoft .NET Framework was used to develop the GUI of the model to provide the client an intuitive experience while interacting with the program. Python3 was used to implement the backend scripts, which aids in the functionality of the GUI application. The major Python3 packages used where pyrebase5 and pyaescrypt. The database and cloud storage were implemented using google firebase.

5.1 User Registration

The user can register using their email id and a password of their choice, At least one digit, one lower case character, and one upper case character must be included in the password to complete the registration process.

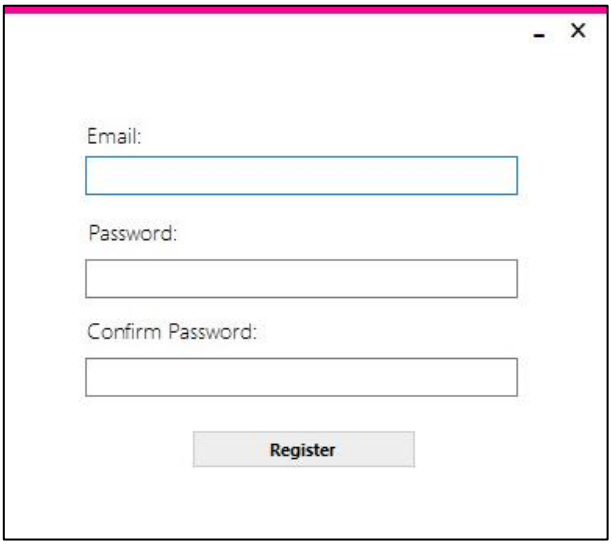

Fig 7: User Registration Page

5.2 User Login

The user can login using existing credentials created in the past. If they forget their credentials they can use the forgot password option which prompts them to enter the registered email id and thus a reset link will be sent to their inbox which will help them to retrieve their credentials.

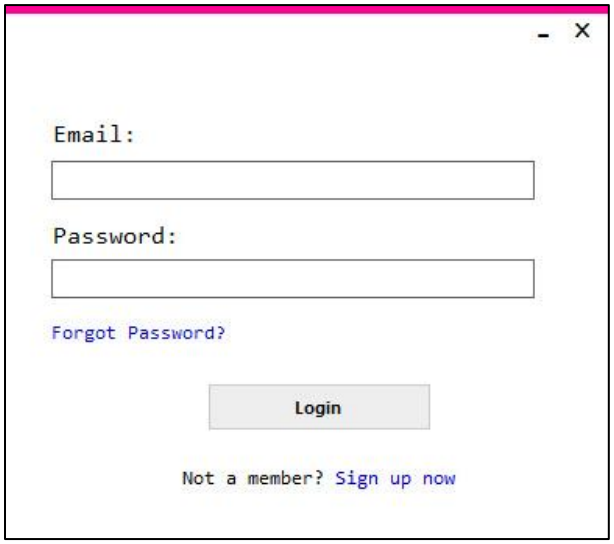

Fig 8: User Login Page

5.3 File Encryption and Upload

The user can browse and select the required file using the browse function, when the file is selected an encryption prompt appears, Figure 9. The user can enter their desired encryption key and confirm the encryption. Once the selected file is encrypted using AES 256 CBC the file is immediately uploaded to the cloud storage.

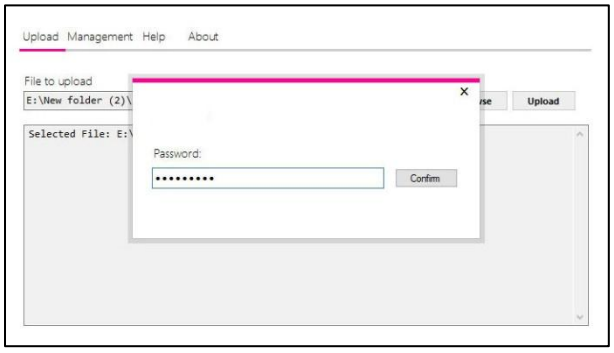

Fig 9: Encryption Prompt

5.4 File Download and Decryption

The user can download files from the cloud storage using the download function. After the file is downloaded a decryption prompt is displayed for the user to enter the decryption key. After successful decryption the file is displayed to the user, if not an error message is displayed.

Fig 10: Download Page

5.5 File Deletion

The user can also delete the files stored on the cloud by selecting the required file and using the delete function.

6. FEATURES OF THE MODEL

- Since the proposed model uses client-side encryption the delay in encryption is minimized.
- The model is immune to brute force attacks making it more secure.
- Theoretically the model is quantum resistant if the key generated is of 256 bits according to Grover's algorithm since the algorithm reduces the brute force attack time to its square root.

7. BENEFITS OF THE MODEL

- Professionals from many sectors are forced to work from home during the COVID-19 pandemic. They could be benefited by keeping their data secure.
- Health care professionals deal with a large number of personal medical records and patient information that must be kept confidential. They can benefit from using this model to secure their patient data.
- Banking professionals can also use the proposed model to encrypt important banking information.

8. CONCLUSION AND FUTURE SCOPE

In this study, security issues with existing models were analyzed and a new model was proposed. AES 256-bits CBC encryption was used for encryption which can withstand existing brute force attacks. Also, the encryption process is from client side so the delay in encryption is minimized.

In spite of the varied benefits, the model can further be improved for its users by implementing it on various platforms and by enhancing exception handling.By employing a key generation function, the encryption can be improved. The limitation of non-retrieval of the data, if the client forgets the decryption key must be also addressed.

9. REFERENCES

[1] M.P Babitha and Babu K. R. R., "Secure Cloud Storage Using AES Encryption," in International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 2016, pp. 859–864.

[2] R Ashalatha, M Vaidehi, "The Significance of Data Security in Cloud: A Survey on Challenges And Solutions on Data Security", International Journal of Internet Computing, Vol, 1, Iss. 3, 2012, pp.3-10.

[3] Lee Bih-Hwang, Kusuma Dewi Ervin, Muhammad Farid Wajdi, et al. "Data Security in Cloud Computing Using AES Under HEROKU Cloud", The 27th Wireless and Optical Communications Conference (WOCC),2018, pp.1-5.

[4] Reema, Princy, and Kumari S. et al., "Security in Cloud Computing using AES & DES," Int. J. Recent Innov. Trends Comput. Commun., vol. 5, no. 4, pp. 194–199, 2017.

[5] Zissis D and Lekkas D, "Addressing cloud computing security issues" Futur. Gener. Comput. Syst., vol. 28, no. 3, pp. 583–585, 2012.

[6] Dave Wallen, "AES Encryption: A Closer Look at Advanced Encryption Standards," Security Boulevard,2020. https://securityboulevard.com/2020/05/aes-encryption-a-

closer-look-at-advanced-encryption-standards/

[7] Peter Johannes Holzer, "The AES-CBC Cipher Algorithm and Its Use with IPsec", HJP,2003.https://www.hjp.at/doc/rfc/rfc3602.html

[8] Mohit Arora, "How secure is AES against brute force attacks", eetimes,2019. https://www.eetimes.com/how-secure-is-aes-against-brute-force-attacks/.

[9] Stallings, W. Cryptography and Network Security, 4/E. Pearson Education.