EasyChair Preprint

№ 9237

Securing the Data in Cloud Using Fernet Technique

Chinni Prashanth, D Bala Sai Teja and V Lavanya

November 3, 2022

Securing the data in cloud using fernet technique

Chinni Prashanth[1], D Bala sai teja[2], Lavanya V*

[1,2] Department of Networking and Communications
SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil nadu, India

ch8254@srmist.edu.in , ds7289@srmist.edu.in, lavanyav@srmist.edu.in

Abstract - A user can request to access or retrieve the data which is stored in the cloud without having direct access to the server computer in a variety of industries, including business, the military, higher education, etc. Governments, businesses, and individual users are actively moving their data to the cloud to cut operational costs. Security is the main issue with cloud-based online data storage, though. Therefore, the goal of the project is to create a cloud-based safe data storage system. There are many ways to address this security issue; in this case, we're using the cryptographic technique of encryption and decryption. The file is uploaded and downloaded using a cryptographic process, which involves encryption and decryption. Data access is made available to privileged users. Here, both the encrypted key and the cipher text will be kept on our local disc according to our paradigm. Once the encrypted key is accurate, cipher text can be decrypted.

Keywords: **Cloud storage, data security, cryptography, access control, encryption.**

I. INTRODUCTION:

The beginning of the information era and an exponential rise in the use of digital computation have both been brought about by the digital revolution. Businesses are growing internationally. This has increased the requirement for data access from any location. Cloud computing can be used in this situation. Storage is one of the most economical cloud computing services, used by both corporations and individuals to offload their large data to unreliable servers. A crucial element of cloud computing is data security. Security is vital, but so is knowing how to handle security for sensitive and non-sensitive data. Confidential information needs to be guarded from unwanted access.

In this study, a system that stores data after encryption is reviewed. Thus, even if a security breach occurred, the attacker would still be able to access encrypted data, maintaining the confidentiality of the data. The user uploads a file through the portal in this system, which encrypts it before uploading it to the cloud. Through the portal, the user can download files from the cloud, which causes the decrypted file to be downloaded to their local computer. Here, the data is encrypted and decrypted using the Fernet method.

The assignment's cryptography portion was completed using the cryptography library, which also provides excellent documentation. The encryption I'll use is called Fernet, and it employs HMAC with SHA256 for authentication and AES in CBC mode with a 128-bit key for encryption. It uses the salt of the computer to initialize vectors. I then looked into typical methods of storing login information, such as username and password, and discovered a tool called pickle that lets you store Python objects and re-load them into programs after they have finished running. This implied that I could use the login and password system whenever I wanted. A key-value pair (dictionary or hash map) is a popular approach to this problem.

The Fernet algorithm makes guarantee that a message that has been encrypted with it cannot be changed or read without the key. An example of symmetric authenticated cryptography is Fernet. For data security and privacy challenges the fundamental difficulty of separating sensitive data with non-sensitive data and permissions has been met, making progress toward addressing concerns about data security and privacy. Through the use of cryptography, the original data are converted into a format that is unreadable. Cryptography can be divided into two types public key cryptography and symmetric key cryptography. This method employs the usage of keys to transform data into a format that is unreadable. As a result, only individuals who have been granted permission can access data stored on cloud servers. Therefore, only authorized users are able to access data on cloud servers.

Data encrypted with a cipher is visible to everyone. Maintaining the appropriate level of data storage security requires administrators and managers of storage systems to perform a delicate act of balancing. They have to take into consideration the three key concerns that are encompassed by the abbreviation CIA: availability, integrity, and confidentiality. They are responsible for preventing sensitive data from being accessed by of unauthorized individuals, ensuring that the data stored in company systems is accurate, and making sure that the information involved is accessible to all employees of the company who have a legitimate need to do so. In the meantime, time, they have to keep a close eye on both the costs involved and the importance of the information they collect. Nobody wants to find themselves in a situation where the cost of their digital storage security systems is more than the worth of the data they are tasked with protecting. However, organizations must also implement security mechanisms that are robust enough so that potential attackers would have to spend more effort and resources breaking through them than the data would be worth in the end.

II. LITERATURE SURVEY

In [1], To ensure data security on a cloud-based server, encryption of sensitive data is done. How to reliably and effectively acquire ciphertext is now the challenge. On the other hand, keyword search (PEKS) is employed in order to extract ciphertexts from clouds without disclosing sensitive information, but the majority of PEKS protocols are unable to fend against a keyword guessing attack (KGA) launched by an unstable cloud server. Meanwhile, the inadequacy of these methods to detect weaknesses results in information leakage. The scalable public-key cryptography with cryptographic reverse firewalls (SPKE-CRF) which is implemented in this is using JPBC (Java Pairing Based cryptography) library. Security research reveals that the SPKE-CRF protocol can fend off chosen keyword attacks (CKA), key generation attacks (KGAs), and algorithm substitution attacks without secure channels (ASA).

In [2], Many academic institutions, governmental organisations, and commercial enterprises are embracing the cloud environment due to its maximum scalability, lowest initial capital investment, and other benefits. Although the cloud environment has many benefits, it also has certain limitations. Data protection is crucial in the context of information technology and cloud computing. Many solutions are used to solve this problem. There is a dearth of in-depth research among the present solutions, making it important to research, categorise, and analyse the significant existing work in order to Determine whether or not these solutions can be used to meet the requirements by evaluating their applicability. The following article compares and examines in-depth the main strategies for transferring and protecting data in a cloud environment. Each specific method is examined in detail, including its role in data protection, potential ground-breaking

In [3], the field of computing, cloud computing has recently grown in significance. In this industry, resources are now simply called up as needed from a variety of cloud vendors rather than first having the infrastructure set up and then being used. For a range of services and data storage, it is also used in many enterprises. Data from the cloud can be requested by users, but many users are concerned about the security of their data. The security concern that most consumers have can be resolved using methods like steganography and cryptography. Cryptographic techniques like DES and AES ensure the data's confidentiality, however often using just one technique may not produce a high level of protection. In this post, we have focused on developing a hybrid cryptographic mechanism that encrypts and decrypts data using a variety of techniques. Cryptographic techniques like DES and AES ensure the data's confidentiality, however often using just one technique may not produce a high level of protection. In this post, they have focused on developing a hybrid cryptographic mechanism that encrypts and decrypts data using a variety of techniques. This proposed system uses the 3DES (Triple Data Encryption Standard) and Blowfish algorithms to offer security. The encryption in this example consists of three parts. Each component is encrypted as necessary using a different encrypted methods and decrypted using different keys. Using 3DES and Blowfish to store the data in cloud server this approach gives users a better data security.

In [4], A fresh strategy for the establishment of public key infrastructure is presented in this article. The public key infrastructure (PKI) has the drawback of requiring the mathematical link in between secret keys to be maintained. The paper makes a recommendation for a whole new PKI system that has addressable parts (PKA). The mathematical connection that exists between public and private keys can be eliminated with the help of the proposed solution, which makes use of addressable cryptographic tables. The use of key aggregation cryptography makes it possible for cloud storage to safely share data. The Advanced Encryption Standard (AES) and other techniques are coupled with the private keys. Using algorithms like the one created by Shor [2], the production of private keys is accomplished without the use of mathematical computations that might be sensitive to quantum computers. PKA moves quickly and uses about 800 CPU clock cycles. The PKA dynamic data encryption strategy was put into place and tested in legacy systems to protect PC-to-PC and PC-to-smart card communication using AES.

In [5], According to statistics from security companies, academic institutions, and governmental agencies, there have been an increasing number of data leak incidents in recent years. Human error is one of the main sources of data loss in a variety of data breach instances. Solutions for identifying accidental sensitive data leaks brought on by human error as well as alerting organizations are available. Screening content for exposing confidential material as it is being stored and sent is a frequent strategy. Such a strategy typically calls for the detecting operation to be carried out in secret.

III. FLOW OF ACTIVITY

A. IMPLEMENTATION:

Data Owner:

User who uploads the data into cloud storage.
User name and password have a crucial role as they are used as a part of key for encryption.
The meta data is store with username and password so these will be used as key for saving cryptographic keys. And also this will act as a barrier between the data storage and user. Though the data is retrieved from the cloud storage the attacker can't access the file as the data is encrypted so to access the attacker must know the username and password. And the application is not a web application the inspect feature is not available to manipulate the code.

Encrypting the data using fernet

This is the major part of the project. Describes the implementation of the encryption and decryption of the data. Encryption of data for various formats of data. Basically, encryption of images takes using the XOR operation but the key is generated using the timestamp which is the core of fernet cryptography. The jpg file which is image converted into the binary format (bin) then doing the XOR operation between the bin file and timestamp. As the timestamp is the encryption key this will be stored automatically after the successful uploading into the storage. If the data is not uploaded because of internet issues then there is no loss of data as the data is encrypted and it will automatically make it available.

Decrypting the data while retrieving

As we are using symmetric cryptography to decrypt the data we need the encryption key so after selecting the file from available files. Then the user, need to enter password then the file is decrypted and downloaded. In case of wrong password, it shows error message and asks to re-enter the password. Even though, the data is downloaded it is in encrypted form. The same XOR operation is done between the data which is in form of binarray and timestamp which is stored as key.

Setup of storage class:

To demonstrate the storage class for cloud AWS S3 storage is used and configured using boto3 module. Boto3 is a Python Software Development Kit (SDK) used to access the resources of AWS based on IAM polices and users.
According to the project requirement the Simple storage service S3 is only accessible to the user and the putObject, getObject, deletObject, viewObject permissions are given for insert, delete, download and view features.

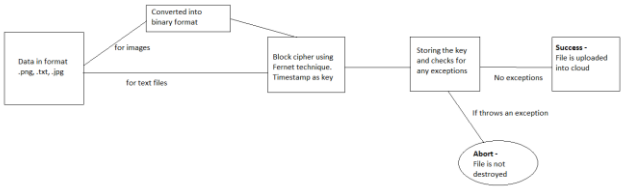

Fig 1: Architecture diagram

The below figure describes the activity diagram of the modules which are implemented and the cryptographic part is described above with format and transaction of activity.
–

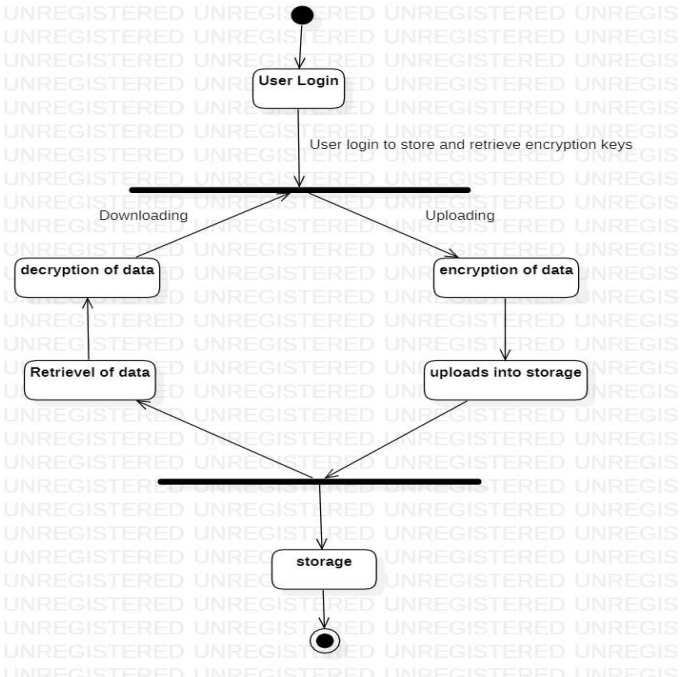

Fig 2: Activity Diagram

IV. SOFTWARE MODULES:

Cryptography:

Python programmers can find cryptographic recipes and primitives in the cryptography package. It should serve as your "cryptographic standard library," according to us. Both Python 3.6 and PyPy3 7.2 are supported.
The cryptographic module is divided into two levels. First one safe cryptographic which is safe and can be implemented easily with importing and making small changes.
Second one is low level cryptographic primitives which is dangerous and can be used incorrectly. They are not easy to implement and have depth knowledge of cryptographic concepts. These can be used by importing hazmat from cryptography.

Pickle library:

For serializing and de-serializing, a Python object structure, use the pickle module in Python. In Python, any object can be pickled in order to save it to disc. Pickle "serializes" the object before sending it to a file, which is what it does. A Python object (list, dict, etc.) can be turned into a character stream by pickingling. This character stream is supposed to contain all the data required to recreate the object in another function.

Fernet Technique:

Using current best practices, Fernet is a system for symmetric encryption and decryption and it authenticates the messages that the recipient may determine whether it has been changed from the version that was initially sent. When creating such a system a novice developer might commit a number of obvious blunders.

Fernet avoids these by supplying a safe method for creating keys (f similar to password). Choosing an effective encryption algorithm AES using CBS mode and PKCS7 padding creating a secure “salt” value at random to strengthen the encryption. The message is signed (using HMAC and SHA256) to prevent tampering. Fernet provides symmetric, or private keys, cryptography, which requires the preservation of a single key that is used for both encryption and decryption. And it is not suitable for large sized files as it has loads the whole buffer into memory at a time.

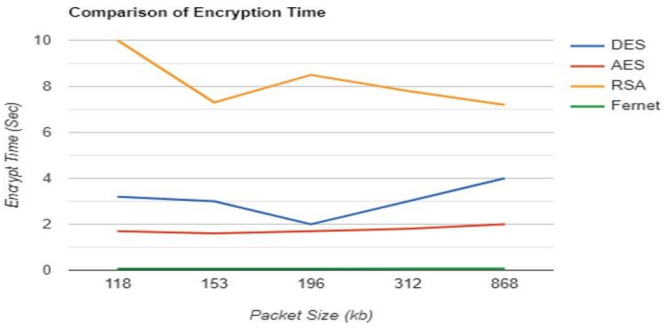

Fig 3: Analysis

Cipher block chaining:

In plaintext pattern hiding, the CBC (short for cipher-block chaining) mode of the AES block cipher performs better than the ECB mode. The initialization vector and the initial plain block (B1) are XORed together before being encrypted in CBC mode to achieve this. Although each new unencrypted component is XORed with the block cipher of the previous block, block chaining is a component of CBC as well. The above is summarized into formula as

Ci = EK (Bi ⊕ Ci-1)

Where Ci-1 is the cipher corresponding to Bi-1 and EK stands for the block encryption algorithm employing key K.

In the above formula let us assume C0 as initialization vector

Similarly, for decryption CBC will be:

Bi = DK(Ci)⊕(Ci-1)

Where K is the key used in the block decryption method DK. For decryption, the same initialization vector (C0) will be applied.

Request module:

It is a standard module in python used to handle HTTP requests. Using this module post, get, status code, JSON response, body, header, content can be managed. URL is used to connect to server.

Boto3 is AWS SDK for python by creating a IAM user with required privileges and policies the request can perform the operations on the resources. It makes easy to integrate with python applications, libraries and scripts .

V. EXPERIMENT RESULTS AND DISCUSSIONS

Before Encryption:

After Encryption:

The above figures are results after the encryption the file becomes unsupported so to make this accessible the user need the key which is stored so the file becomes normal.

ECB and CBC are two of the many different modes of operation that are accessible for block ciphers. Because every one of these approaches has a unique set of benefits and drawbacks, selecting the approach that is most suitable for the effort is contingent on its specifications. The ECB and CBC modes are both examples of modes that provide privacy, whereas other modes, such as the Galois Counter Mode (GCM), provide both privacy and integrity of data.

It is best to use the CBC mode whenever possible rather than the ECB option. As was explained before, the ECB approach reveals information about the plaintext since similar plaintext blocks result in identical ciphertext blocks. As a consequence of this, the technique exposes details about the plaintext. Because a ciphertext must never disclose details about the plaintext that was used to build it, using ECB mode is risky and should be avoided at all costs.

VI.CONCLUSION AND FUTURE WORK

A better level of security is provided by Fernet, which also offers the best cryptographic module for Python and is simple to deploy and work with the keys. The keys were obtained from the timestamps placed next to each when it comes to encrypting the data, a few seconds can make a significant difference.

And Pickle saves these keys in the form of Python objects, which can then be used at a later time after the objects are converted into binary. However, reading from and accessing the pkl file does not appear to be possible without the script.

REFERENCES:

[1] Yunyang zhou , Zhebin Hu and Fagen Li **,"**Searchable Public-Key encryption with cryptographic
reverse firewalls for cloud storage" ,IEEE,2021.

[2] Isha gupta , Ashutosh kumar singh , Chung-nan lee and Rajkumar Buyya,"Secure Data Storage and Sharing techniques for data protection in cloud environments",IEEE,2022.

[3]Vivek Sharma, Abhishek chauhan , harsh saxena ,and Sulbah Bansa, "Secure File Storage on cloud using Hybrid Cryptography",IEEE,2021.

[4]Bilal Habib, Bertrand Cambou, Duane Booher , Christopher Philabaum, "Public Key Exchange scheme that is Addressable (PKA)", IEEE , 2017.

[5] Xiaokui Shu , Dandeng Yao and Elisa Bertino, **"**Privacy-Preserving Detection of sensitive date exposure",volume:10 issue:5 , IEEE,2020.

[6] D.Bhole,A.Mote and R.Patil , " A new secutrity protocol using cryptography algorithms",International journal of computer sciences and engineering, vol. 4 , no.2,pp. 18-22, 2016

[7] B.Bindu, K.Lovejeet and L.Pawan , "Secure File Storage In Cloud Computing Using Hybrid Cryptography Algorithm", International Journal of Advanced Research in Computer Science, vol.9 , no.2,2017

[8] C.Biswas, U.D.Gupta and M.M.Haque, “An Efficient Algorithm for Confidentiality Integrity and Authentication Using Hybrid Cryptography and steganography”, International Conference on Electrical Computer and Communication Engineering.,pp.1-5,2019.

[9] M.Z.Abdullah and Z.J Khaleefah, “Design an d implementation of a hybrid cryptography textual system”, International Conference on
and Technology , pp.1-6, 2017.

[10] “Scheduling of Scientific Workflows in Cloud with Replication”

[11] “Secure File Storage using Hybrid Cryptography” Putta Bharathi; Gayathri Annam; Jaya Bindu Kandi; Vamsi Krishna Duggana; Anjali T.

[12] “An Approach to Hybrid Cryptography on Cloud EnvironmentsMr. Rohit Barvekar, Mr. Shrajal Behere, Mr. Yash Pounikar, Ms. Anushka Gulhan, 2018

[13] Ping, Z. L., Liang, S. Q., & Liang, L. X. (2011). RSA Encryption and Digital Signature. 2011 International Conference on Computational and Information Sciences.

[14] Neha Shrikant Dhande, FOG COMPUTING:

REVIEW OF PRIVACY AND SECURITY ISSUES, International Journal of Engineering Research and General Science Volume 3, Issue 2, March-April 2015.

[15] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, ‘‘(Leveled) fully homo-morphic encryption without bootstrapping,’’ ACM Trans. Comput. Theory, vol. 6, no. 3, pp. 1–36, Jul. 2014.

[16] N.Attrapadung and H. Imai, ‘‘Attribute-based encryption supporting direct/indirect revocation modes,’’ in Proc. IMACC, Cirencester, U.K.,Dec. 2009, pp. 278–330.

[17] Z. Brakerski, Y. T. Kalai, J. Katz, and V. Vaikuntanathan, ‘‘Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage,’’ in Proc. IEEE 51st Annu. Symp. Found. Comput. Sci.(FOCS), Las Vegas, NV, USA, Oct. 2010, pp. 501–510.

[18] A. Sahai and B. Waters, ‘‘Fuzzy identity-based encryption,’’ in Proc.24th Annu. Int. Conf. Theory Appl. Cryptograph. Techn. Berlin, Germany:Springer, 2005, pp. 457–473.

[19] Shaikh, S., & Vora, D. (2016). Secure cloud auditing over encrypted data. 2016 International Conference on Communication and Electronics Systems (ICCES).

[20] Mr. Gajanan N. Tikhe, Mr. Yogadhar Pandey, “A Secure Scheme to Avoid Worm hole Attacks in Ant based Adaptive Multicast Routing protocol for MANET”, IFRSA’s INTERNATIONAL JOURNAL OF COMPUTING (IIJC) Volume 2, Issue 1, ISSN (Print):2231:2153, ISSN (Online):2230:9039, Jan 2012.

[21] Neha Shrikant Dhande, FOG COMPUTING: REVIEW OF PRIVACY AND SECURITY ISSUES, International Journal of Engineering Research and General Science Volume 3, Issue 2, March-April 2015.

[22] Kumar, A., Lee, B. G., Lee, H., & Kumari, A. (2012). Secure storage and access of data in cloud computing. 2012 International Conference on ICT Convergence (ICTC).

[23] Neha Shrikant Dhande, “FOG COMPUTING: REVIEW OF PRIVACY AND SECURITY ISSUES”, International Journal of Engineering Research and General Science Volume 3, Issue 2, March-April 2015.