COST-EFFICIENT DATA PRIVACY PROTECTION IN MULTI CLOUD STORAGE

Artem Matveev

Buryat Institute of Info communication (branch of) Siberian State University of Telecommunication and Information Science, Ulan-Ude,
Republic of Buryatia, Russia

ABSTRACT

Data privacy in the cloud is a big concern for all of its users, especially for public clouds. Modern trends in studies utilise multiple clouds to achieve data privacy protection. Most of the present studies focus on business-oriented solutions, but current study aims to create a solution for individual users which would not increase the cost of ownership, and provide enough flexibility and privacy protection by combining password protection, key-derivation, multilayer encryption and key distribution across multiple clouds. New design allows to use single cloud to store protected user data, meanwhile use free plans on other clouds to store key information on others and thereby does not rise a cost of the solution. As a result, proposed design gives multiple layers of protection of Data Privacy while having a low cost of use. With some further adaptation it could be proposed as a business solution.

KEYWORDS

Multi Cloud Storage, Privacy Protection, Password, Key Distribution, Cloud Data Security.

1. INTRODUCTION

Increasing demands in processing data, quick service deployments with high-availability and at low costs resulted in forming a cloud computing model, also known as clouds, where separate users are sharing common resources, maintained by the cloud provider. Cloud storage is one of the services provided by cloud providers which allows users to store data on the provider's servers.

As the amount of data increases [1], [2], more and more data ends up uploaded to the cloud storage. Despite cloud providers does not provide direct statistic of data volume inside cloud storages, trend could be seen through growth of both revenue and user base, like in Drop box reports: [3], [4], [5], [6], [7], or can be found in analytics reports, such as [8].In many cases, especially in public clouds, the cloud provider is an independent organisation or person which means that data uploaded to the cloud is maintained and accessible not only by the end user itself, but by the cloud provider and its affiliated organisations or persons. These raise concerns over data privacy in the clouds. Past occurrences [9], [10], [11], [12], [13], [14], [15] have shown that this concern is not groundless, and some incidents with data leakage, business espionage and even government spying over the data confirms that.

The most effective solution to maintain data privacy is local side encryption performed shortly before data would be send into cloud storage. Meanwhile, modern times have shown that data located in cloud storage is required to be accessible at any time from any device. This is

David C. Wyld et al. (Eds): DMML, SEAS, ADCO, NLPI, SP, BDBS, CMCA, CSITEC - 2022
pp. 67-81, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120706

especially stimulated by the Bring Your Own Device (BYOD) trend in businesses where employees can access data from their own mobile devices. On the other hand, individuals or small business users can have only a single device to access the data and still want to maintain their data privacy. It also needs to be considered that this device can be stolen, lost or severely broken (further referred to as lost).However, encryption requires an encryption key. As a result, the questions are raised: where to locate a key; how to keep it secret and perform its safe sharing and recovery in cases of loss.

The first simple solution is to use a password-based key generation, for example, described in [16]. This approach solves the issues where data would be inaccessible from another device or in case of device loss. But from the other point of view it needs to be considered that this way opens attack on brute forcing passwords (including usage of password dictionaries) and entire crypto strength fully relies on the password. Furthermore, it needs to considered that typically users are not using strong passwords or reusing their passwords so that making such attacks are a real issue. Finally, it also needs not to be forgotten about possible social engineering attacks on the end user. Altogether, these facts making this approach vulnerable to the end user behaviour.

Another solution is to save the encryption key on some end user's device such as a computer or smart-card. This way increases the cost of infrastructure and limits the list of devices from where access is available. Also, in case of device loss the data becomes unrecoverable. On the other hand, lack of proper infrastructure or improper actions with keys can lead to encryption key leakage [17]. As a result, this way is more oriented for some business applications rather than for individual users.

Next generation solutions no longer rely on single cloud provider and use the "divide and conquer" idea, but applied to data privacy security. The basic concept is to slice data into separate chunks and store them in multiple cloud storages from different providers. This opens a new promising paradigm in data security and privacy – multi cloud storage [18]. Since multiple cloud storages from different providers are used, one provider can no longer have full access to the data and that makes the end user the only person who is able to get all the data. From the intruder's perspective, access to the full data is also becoming problematic, with the exclusion of MITM and end-user device attacks, they need to gain access to several cloud providers. But this basic concept is still vulnerable to data guessing and is still able to disclose some part of the original data. Although those issues could be resolved by using data encryption, there arises 2 more issues which need to be resolved: (i) how scheme stores/derives the encryption key; (ii) the rising cost of the overall solution, due to the requirements in applying for multiple subscriptions.

In order to answer those issues and guarantee data privacy in cloud storage, this paper proposes a new design of using a multi-cloud environment and encryption which on one hand will utilise a single cloud to store the actual data and use password protection, but on the other hand involves a multi-cloud paradigm to prevent password brute force attacks. The proposed design provides multiple levels of protection while having low cost of use, because data is actually stored in a single cloud. This work mainly aims at maintaining privacy of individual or small business users, although with some further adaptation it could be proposed as a business solution. That achieved by combing existing well-known solutions (encryption, key-derivation) in specific order boosting security by using multiple clouds to store keyed information. New scheme compared with existing solutions from perspective of complexity crypto operations and from perspective of resistance on existing threads.

The remainder of the paper is formed as follows. Section 2 describes the overview of the related work in the field. Section 3 describes proposed scheme. Section 4 discussed implementation

aspects and performing threat and complex assessments. Section 5 concludes the report and future work.

2. LITERATURE SURVEY

A performed literature survey has shown a lot of research studies in a field of data security and privacy in a cloud throughout the past 2 decades. Although research studies have progressed especially in introducing using multiple clouds to reach the goals of data security and privacy, they avoid scenarios for personal usage purposes ([18], [19], [20], [21], [22], [23]). The survey results shown that only two studies have aims in design to reach individual users and another one has the perspective of being used as a possible solution, but with different intentions.

Studies [24], [25], [26] have performed state-of-art surveys and outlined possible threats, issues and ways of protecting data. Possible threats which were listed in those works are the following: Password cracking or Brute Force; Inconsistent Use of Encryption; Catastrophic Hardware Failure; Malware; DDoS; Man in the middle attack; Data leakage/Side channel attacks; Data Disclosure;

In [25], [26] also highlighted main aspects of data protection in clouds such as: Data Confidentiality; Data Integrity; Data Availability; Non-repudiation of Actions; Fine-Grained Access Control; Secure Data Sharing in Dynamic Group; Leakage-Resistance; Complete Data Deletion; Privacy Protection.

Study [26] declares Data Confidentiality as the ability to prevent the active attack of unauthorized parties on users' data, and ensure that the information received by the data receiver is completely consistent with the information sent by the sender; Data integrity as the reliability of the data, that is, the data cannot be arbitrarily tampered with and replace; Data Availability as Data availability emphasizes that data can be accessed normally at any time and Privacy protection as the ability to guarantee sensitive data protection under curious adversaries and malicious employees of cloud service providers. [25] describes Non-repudiation of Actions as (aspect which) ensures that neither party will be able to deny the occurring transaction.

A discussion about possible solutions (in-terms of fully backwards recoverable data) to these threats in [24] - [26] contains the following: use of strong passwords; hierarchical role-based access control; data encryption; using security level and data classification; tokenisation; Identity-Based Encryption; Attribute-Based Encryption; Homomorphic encryption; Searchable Encryption. Despite this, from a mathematical perspective, strong passwords represent a good enough solution and have the ability to protect data, [24] states that choosing and the proper usage of such kinds of passwords became not just a technical issue, but a behaviour one.

Other solutions mainly propose ways of how to perform Secure Data Sharing and building access systems. And the last is raising questions about the ability to find documents through performing confidential cloud searches.

Study [1] demonstrates one of the possible scenarios of usage – data collection from sensors located on/in factory's equipment. Individual users can experience similar scenarios of use with heavy growth of the Internet of Things. But this proposed solution requires installing an intermediate server which could not being suitable for individual usage.

Study [27] proposes a way of building cloud storage, but it also includes a proposal of using "boot code" technology, which uses a password and file name to generate an encryption key.

However, this method is equally vulnerable as just using passwords, because the filename is not hidden in this case.

Studies [18], [19], [20], [21], [22], [23] use a new paradigm – multiple clouds. Some of the works are just slicing files and then storing chunks in different clouds [20], the rest of the works include encryption of data chunks. Study [19] described an implemented prototype for eGovernment purposes in a conference. As one of suggestion was proposed to look into the field of mix networks and the security modelling used there. But none of the work is responding to the question about how shall the encryption key be located.

Study [28] addresses an issue about availability of such a solution for individual users. The basic idea is pretty similar as outlined in [18] - [23] – original file encrypting, splitting, compressing and storing in multiple clouds, but small pieces of the beginning and the end of files are stored locally at the end user device. This approach allows protecting data in the cloud, because in cases of using streaming symmetric ciphers there would no ability to properly decrypt a file, but this way placed restrictions on data accessibility and in cases of losing users' devices the data will be lost.

Study [29] suggests using a single location for encrypted data and later sharing information of the key by using Shamir's secret sharing and later storing it in different locations. But this work is not aiming at providing an end solution for individual users. Also, study [30] stated that this scheme is partially trusted and has a high computation cost.

Study [31] proposes using biometry, an identity server and user's auxiliary devices in order to protect the encryption used for encrypting backups from end-users' mobile devices. The main aim of the scheme is to provide safe backups of users' IDs and payment information and later the ability to restore them in case the device is lost.

Study [32] discusses issues of providing secure access to the file and meanwhile changing the encryption key. This solution is inventing proxy-servers which are performing re-encryption.

Study [33] provides the opportunity to protect secrets with distribution parts of the secrets in several servers, but the algorithm binds to private and public keys which is not suitable for public clouds where servers are not maintained by the end user.

3. PROPOSED DESIGN

3.1. Design Overview

Data security and privacy, as it was shown in 2. Literature Survey section of this work, are widely discussed. Different studies are proposing different approaches to cover different aspects of data protection. But as it was shown, mainstream studies are concentrated on providing business-oriented solutions. Some of them have potential issues with data availability due to introducing additional intermediate or end-users' servers which could not be as scalable as a cloud provider's infrastructure. Those approaches can also increase networking round-trip times due to ineffective network routing.

This work proposes a fully-multi-cloud solution where the password is left outside of the cloud and must be remembered by user. Meanwhile, protection of end users' privacy is boosted by using a multiple cloud solution. From another perspective, the current proposal only requires a single cloud to store protected data.

In addition, with proper configuration, current design allows protection against a broken encryption algorithm by allowing usage of several independent algorithms in a row.

3.2. Design Requirements and Notation

This section describes a list of features and requirements for the proposed scheme and introduces notations used later in the study. The proposed scheme shall:

- Use single cloud storage to store the main data;
- Use multiple cloud storages to enhance security;
- Provide the ability for the end-user to access data from new devices only by specifying a password and granting access to all clouds;
- Provide the ability to choose different encryption algorithms in order to protect data.

Used notation is described in Table 1.

3.3. Initialisation and Input Data

This section covers the initial and input data required for the scheme. To begin its operation, the scheme requires to be specified:

Table 1. The notation

Symbol	Description
P	Payload to protect
S_j	Cloud Storage
T	Amount of cloud storages
M	Master password
N	Amount of encryption layers
E_i	Symmetric encryption algorithm for protecting payload
R_l	Symmetric encryption algorithm for protecting key gamma sequences[1]
K_i	Session encryption key used in payload encryption
K_M	Encryption key based on master password
G_j	Random transformation gamma sequence[1]
G_{K_i}	Random key-deriving gamma sequence[1]
G_{C_i}	Transformed G_{K_i} with set of G_j
$B_i\ (G, G)$	Gamma-sequence reversible blend function
X_i	Key-extraction function from gamma sequence[1]
C_{P_i}	Encrypted payload
C_{K_i}	Encrypted gamma sequences[1]
A_i	Salt
$h(x, A)$	One-way password transformation function

- P
- M
- List of S_j with size of T
- List of E_i with size of N
- List of R_l with size of $N + T$
- $h(x, A)$

[1]In this study Gamma Sequences is considered as a random sequences of bytes.

- List of B_i (G, G) with size of N
- List of X_i with size of N

Although the list contains more than just payload, password and cloud storages, the rest of the things could be implemented as settings and could be shipped with factory-prepared values. As a result, for the end user it would not be mandatory to specify those values.

For the first launch, the scheme is also required to randomly generate G_j and .G_{K_i}. In other cases, the system needs to extract and decrypt C_{K_i}. This step is described later in Key Information Decryption and Recovering section.

All blend functions must be reversible, i.e. if $X = B(Y, Z)$ then $Y = B\ (X, Z)$ shall be true.

3.4. Protecting Payload

Payload protection basically represents layered encryption of the original payload as shown in Figure 1. The steps are performed as present the equations 1-3 by their numbers.

1. Extracting encryption keys K_i from key-deriving gamma sequences G_{K_i} with extraction function X_i, as in equation 1.

$$K_i = X_i\left(G_{K_i}\right), i \in [1, N] \tag{1}$$

2. Encrypting the payload P several times to form layered encryption with extracted keys K_i with encryption function E_i, as in equations 2 and 3 respectively.

$$C_{P_1} = E_1(P, K_1) \tag{2}$$
$$C_{P_i} = E_i\left(C_{P_{i-1}}, K_i\right), i \in [2, N] \tag{3}$$

3. Store the encrypted payload C_{P_N} to the main cloud storage S_1.

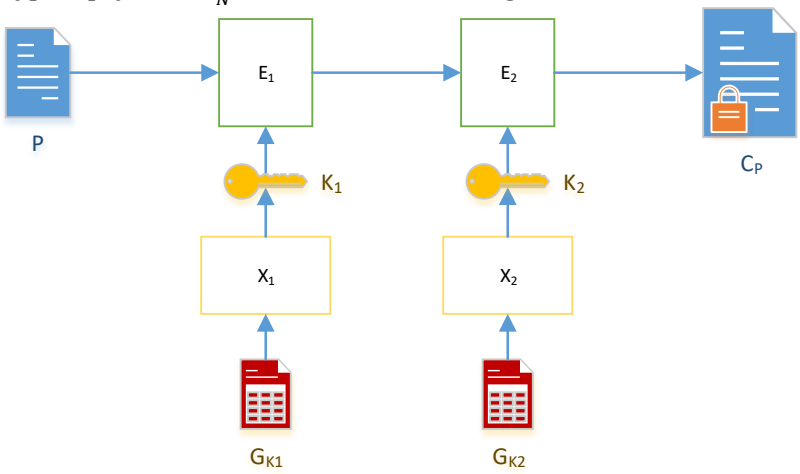

Figure 1. Encryption sequence flow

3.5. Protecting key information

Key information is protected by (i) blending them together and (ii) encrypting it with a master password given by the user. Later this key information could be stored into different cloud storages. The part of the scheme is illustrated in Figure 2.

1. Blend key-deriving gamma sequences G_{K_i} with a transformation gamma sequences G_j with blend function B, as in equations 4 and 5 respectively

$$G_{C_i} = G_{K_i}, \forall\, i \in [1, N] \tag{4}$$
$$G_{C_i} = B\left(G_{C_i}, G_j\right), \forall\, j \in [1, T] \tag{5}$$

2. Encrypt transformed key-deriving G_{C_i} and transformation gamma sequences G_j with the master password M, one-way password transformation function h and encryption function R_l, as in equations 6 and 7 respectively.

$$K_{M_l} = h(M, A_l) \tag{6}$$
$$C_{K_l} = R_l\left(G_l, K_{M_l}\right) \tag{7}$$

where l is either $j \in [1, T]$ or C_i, $i \in [1, N]$; the salt A_l is a randomly generated sequence of bytes, which, in the current scheme, is different for each gamma sequence; K_{M_l} is gamma's sequence's generation's encryption key based on password M and salt A_l.

3. Store encrypted gamma sequences and their salts into different clouds S_i.

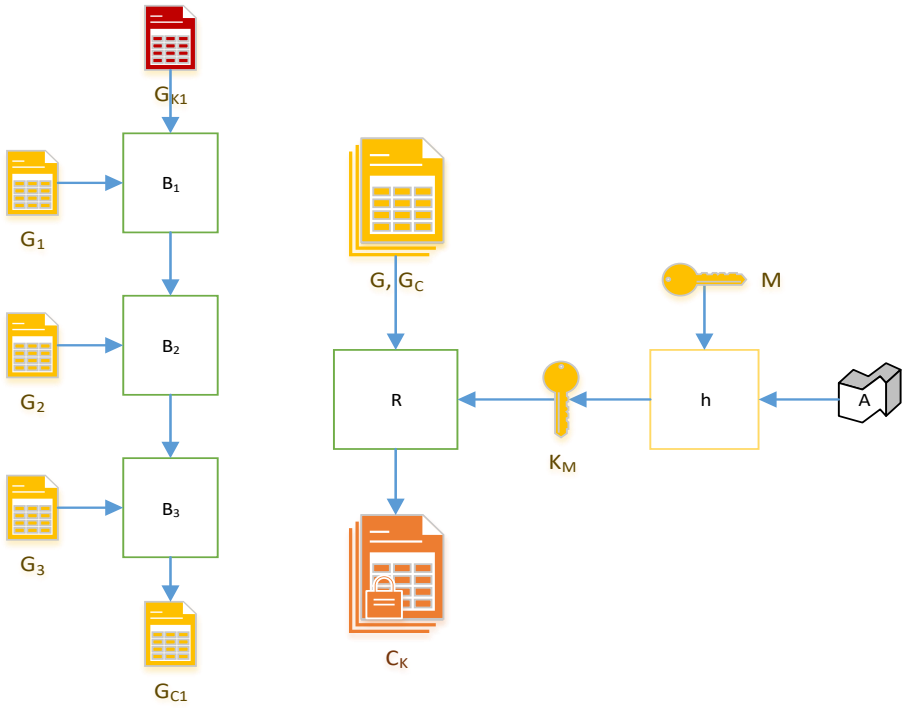

Figure 2. Gamma Sequence transformation and encryption flow

3.6. Key Information Decryption and Recovering

Keys recovering and decryption basically represents backward action presented in equations 4-7. The process of key reconstruction of following steps:

1. Read encrypted C_K and salt A from the cloud storages.

2. Generate an encryption key K_{M_l} from the master password M for each gamma sequence G_l as described in equation 6.

3. Decrypt gamma sequences G_l with given keys K_{M_l} by encryption function R_l as in equation 8.

$$G_l = R_l\left(C_{K_l}, K_{M_l}\right) \tag{8}$$

4. Perform backward blending with the transformed gamma sequences G_{C_i} with a transformation gamma sequences G_j by blend function B to get key-deriving gamma sequences G_{K_i}, as shown in equations 9-10.

$$G_{C_i} = B\left(G_{C_i}, G_j\right), \forall\, j \in [T, 1]\,, i \in [1, N] \tag{9}$$

$$G_{K_i} = G_{C_i} \tag{10}$$

3.7. Payload Decryption

Payload decryption simply consists of (i) loading the encrypted payload from the cloud; (ii) extracting encryption keys by equitation1 and (iii) performing decryption in reverse order for each E_i.

4. DISCUSSION

4.1. Implementation Aspects and Considerations

Some security aspects of the scheme depend heavily on the chosen implementation details and this section intends to highlight possible facts and aspects which need to be considered for the scheme implementation.

The best suited symmetric algorithm mode for this scheme is streaming mode, such as CBC mode at least [34]. The ECB mode is not recommended to use due to its ability to contain vulnerabilities and because it can lead to possible information disclosure [35]. The cryptography strength is dependent on the strength of the strongest used algorithm.

As function h could be considered function PBDKF2 which makes brute forcing the password hard due to its high computational cost. This function is required only in cases of either encryption or decryption keyed information, but does not directly involve generating keys for payload encryption and therefore has little computational cost impact in terms of payload encryption.

Blend function B could just be XOR. Since gamma sequences are random, XOR would be enough to provide a proper XOR cipher scheme.

Extraction function X could be a function which just slices the first bytes from key-deriving gamma sequences, although it could be any other function, including PBDKF2. Since keys are static for all payloads they would be needed to be extracted only once and this shall not significantly increase computational burden.

Also, for the purposes of increasing unpredictability, cipher Message Authentication Code (MAC) could be considered to discard and, instead, implement one's own MAC at the top level before the encryption process starts. This increases the difficulty to guess encryption keys for each layer independently and will require decrypting all layers in order to confirm that all keys and the decrypted payload are valid, meanwhile it still confirms data integrity. As an alternative approach to implementing own MAC, the MAC from the first algorithm could be left in encrypted stream, while the rest of them are discarded.

Meanwhile for gamma sequences, the MAC could be omitted and instead of it, a test file could be used which would be encrypted with multi-layered encryption as a usual file. This makes it impossible to guess a derived key from one stolen set of gamma sequences. The intruder would have to combine all sequences, generate $N + T$ gamma keys, decrypt all sequences, generate N payload keys, decrypt the test payload and check it in order to confirm success.

An additional measure of increasing unpredictability could be considered using randomised start indexes in gamma sequences, which makes it hard to know which section of bytes need to be targeted. Example of such approach shown in

Figure 3.

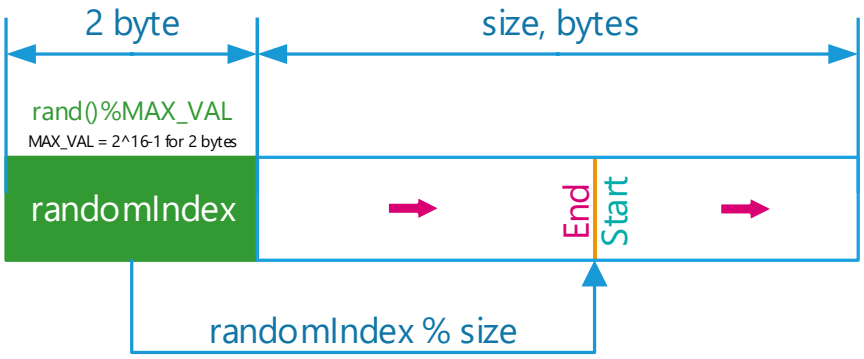

Figure 3. Gamma sequence container with randomised start index

Gamma sequences could be cached in end user devices which reduces risks that this information could be stolen. Additionally, this information on end user devices could be protected by biometric sensors.

The other measure which could be applied to encrypted key-deriving gamma sequences is to avoid/deprioritise using main cloud to store them.

4.2. Threats and Data Protection Aspects Analysis

Current section of present study intends to analyse threats and data protection aspects discussed in this study. Meanwhile it also needs to be considered that providing security at end-user device is out of scope for current research study and this study assumes that end user devices is secure.

4.2.1. Password Brute Force

As was discussed earlier some of the solutions are vulnerable to password brute forcing. It is obvious that full password brute forcing is complex and typically impossible with good length and alphabet, but using password dictionaries makes this task much easier to do. In this case, even if the password was brute forced, the intruder needs to collect all gamma sequences from different cloud storages in order to extract all encryption keys.

Also section 4.1. Implementation aspects and considerations proposes an implementation technique where just owning encrypted keys on the one hand gives no ability to brute force passwords, and, in another significantly slows that process by proposing to use [16] and different salts for each key. Therefore, from this perspective of this threat, the scheme provides Data Confidentiality.

4.2.2. Inconsistent Use of Encryption

The proposed scheme itself could contain such vulnerability, but it was discussed in section 4.1. Implementation aspects and considerations. If proposals would be implemented, those issue would be mitigated.

Furthermore, this scheme allows mixing different algorithms which enables taking the advantages of each algorithm and in case of algorithm cracking still provide security for its content

Therefore, from this perspective of this threat, the scheme provides Data Confidentiality.

4.2.3. Catastrophic Hardware Failure, DDoS

In terms of hardware failure, studies consider that cloud storages typically have strict policies [36], [37], [38] about good hardware redundancy, and most of them store data across several data centres, therefore it very unlikely to that cloud storage will have such enormous failures. Besides, it needs to be acknowledged that individual users typically use a single cloud to store their data and this way does not introduce additional risks, especially if the key cache scheme was implemented on end users' devices providing the ability to recover them -or- end users' can duplicate some keyed information inside cloud storage groups (instead of saving keyed information directly to the cloud – save it into group, like RAID disk groups).

Failure of intermediate communication lines also does not introduce more risks in data loses. Most of them are temporary and won't cause big issues for individual users. Meanwhile, permanent connection loss to keyed information could be mitigated with same ways as cloud storage failure.

The most likely scenario of catastrophic hardware failure would be the end-user's device's unrecoverable inoperability for example device loss. In this case, the user would be able to access their data by accessing their cloud storage accounts and entering their master password. From this perspective, this scheme provides the end-user high Data Availability, even in cases of device loss or an immediate need to access from a new device.

4.2.4. Man in The Middle Attack

Currently all communication between end-user and the cloud storage provider are secure, typically by using the TLS protocol. From this perspective, this study is not considering that a MITM attack could have taken place near the end-user's device.

A MITM attack could be performed inside a cloud provider's data centre, but this does not provide any advantages for the intruder since they need to access keyed information across different cloud providers thereby continuing to provide Data Confidentiality.

4.2.5. Data Leakage/Side Channel Attacks and Data Disclosure

As discussed above, to actually provide an intruder with any advantages of stolen data they need to access several cloud storages. Meanwhile it's very unlikely that several cloud data storage providers will leak information at the same time and it has the same applications on Data Confidentiality as discussed above.

Also, post-incident mitigation measures could be applied – generating new gamma sequences and re-encrypt all content in the background in order to protect it with new keys.

4.2.6. Malware

Malware could be on either the cloud provider's side or end-user device.

Cases of malware on the cloud provider's server are similar to sections 4.2.4. Man in the middle attack and 4.2.5. Data leakage/Side channel attacks and Data Disclosure a scenario that is very unlikely to occur at the same time in different clouds. Cases of malware on the end-user device is out of scope for this study and relates to more specific security tools such as anti viruses.

4.3. Assessing Scheme Complexity

For the purposes of assessing scheme complexity, hereinafter it will be assumed that: (i) proposals from "4.1. Implementation aspects and considerations" are implemented; (ii) single password test iteration is a pair of one key-deriving and decryption operations; (iii) single key test iteration is a decryption operation. In opposition to the current scheme used, simple scheme consists of: (i) key-deriving from the password; (ii) decryption; (iii) MAC/integrity check. Both schemes are equally likely to use same key-derivation and decryption functions and, hereinafter, it is assumed that they are using the same set of functions for each layer independently (if applicable). This section is looking into scheme complexity from two perspectives: (i) password brute forcing; (ii) key brute forcing.

4.3.1. Password Brute Force Complexity

This assessment calculates the amount of password test iterations I which needs to be done to traverse through the entire list of passwords with length equal to $L.L$ is an input value and its calculations are out of scope for this study.

It is obvious that simple schemes take one password test iteration per single password and the overall amount can be found as in equitation 11.

$$I = L \tag{11}$$

The proposed design takes $N+T$ password test iterations to decrypt gamma sequences; N iterations to decrypt payload (if consider B and X functions as a key-deriving function) per single password. The overall amount of password test iterations can be found as in equitation 12. As the proposed design takes $2N+T$ times more iterations. Because of the possibility to choose different algorithms, the actual taken time could be more than that, but, in the worst case, it would not be less than $N+T+1$ times more. This time can be improved by performing Q layered encryption for gamma sequences, where each single layer will have its own salt, presented in equitation 13.

$$I = L(2N+T) \tag{12}$$
$$I = L(N[Q+1]+QT) \tag{13}$$

With an example setup of $T=3$ clouds and $N=2$ layers of encryption, password brute forcing will take 7 times more password test iterations than simple scheme and for $Q=2$ layered encryption of gamma sequences it will be 12 times more iterations.

It is worthwhile to say that this advantage is possible by implementing the following proposals: custom implementation of MAC at top level; removing MAC from gamma sequences. Without them, the complexity could be decreased as specified in equitation 11, but not less than that.

4.3.2. Key Brute-Force

This attack is much harder to be performed, because of the amount of combinations, but due to key information distributed across different cloud services it could be hard for the attacker to get all the necessary information in order to perform a password based attack. In this assessment, the amount of key test operations U would be calculatedwith overall key amount W.

Simple scheme takes one key test operation per each key. This is shown in equitation 14.

$$U = W \tag{14}$$

The proposed design offers N layered encryption and requires to individually pick an encryption key for each encryption level. Also, it need to be considered that it is unlikely that 2 random keys would have identical values (for example, the probability of 2 identical random keys for AES-256 is equal to $P(K_1K_2) = \left(\frac{1}{2^{256}}\right)^2 = \frac{1}{2^{512}}$) and thereby it can be considered that keys are not reused. As a result, the overall amount of iterations would be calculated as k-permutation of n and shown in equitation15.

$$U = A_W^N = P(W,N) = \frac{W!}{(W-N)!} \tag{15}$$

With an example setup of $N=2$ layers of encryption this way will have $W-1$ times more key test operations in order to reach content.

4.4. Known Limitations of the Scheme

The proposed scheme significantly boosts Data Privacy protection inside cloud storages, there exist some limitations of the scheme:

- Requirement of accessing to, at least, 2 cloud storages;

- Scheme itself does not offers resilient against access loss of one cloud, but mitigation strategy was being briefly discussed;
- File search became much harder due to computation cost and network delays;
- Inability to share data with another use.

5. CONCLUSIONS

This work showed that Data Confidentiality in the cloud storages is still experiencing lack of solutions for individual or small business users. To resolve this issue, current study proposed new scheme which is offering a solution for protecting user's data privacy, while not increasing the cost of such protection and utilising advantages of the multiple cloud paradigm. This solution uses combination of well-established security techniques, such as password-based key deriving, symmetric encryption functions, meanwhile involving multiple clouds to place keyed information across them in order to enforce privacy protection and as a result offer multiple levels of Data Privacy protection in the cloud storages. Performed analysis of possible threats and data protection aspects has shown that the scheme offers high levels of protection and mitigates listed threats; analysis of scheme complexity has shown improved complexity against brute force attacks. Also, the study proposes some considerations for possible implementations which can further enhance protection.

Although, this study proposes a ready-to-use scheme and some possible enhancement was discussed earlier in the paper, there still exists the question of implementing a group sharing scheme to allow users to safely share their files.

ACKNOWLEDGEMENTS

I would like to thank a few people who helped me to accomplish this work. First my thanks to my supervisor (Ivan Nechta), who got interested in my idea and work and provided feedback on this paper. Big thanks to my friends from Canada (Jordin McEachern) and UK (Ahmed Elakehal) who reviewed and helped me fix my English misspelling and errors. Also, big thanks to my cat, Cisco. She was sitting around me during my entire first and third days of writing this paper.

And special thanks for the Organizers of ADCO for providing the opportunity to publish this work.

REFERENCES

[1] J. Liu, C. Yuan, Y. Lai and H. Qin, "Protection of Sensitive Data in Industrial Internet Based on Three-Layer Local/Fog/Cloud Storage," Security and Communication Networks, vol. 2020, p. 2017930, 04 2020.

[2] B. Marr, "How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read," 21 05 2018. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/.

[3] Dropbox, Inc., "2018 Press Release," Q4 2018. [Online]. Available: https://dropbox.gcs-web.com/news-releases/news-release-details/dropbox-announces-fourth-quarter-and-fiscal-2018-results.

[4] Dropbox, Inc., "2019 Press Release," Q4 2019. [Online]. Available: https://dropbox.gcs-web.com/static-files/28e269b2-3442-43bb-b635-53f72e3d26f3.

[5] Dropbox, Inc., "Q4 2020 DBX Investor Presentation," Q4 2020. [Online]. Available: https://dropbox.gcs-web.com/static-files/37497899-90c7-44a1-91da-be8d1c6ed333.

[6] Dropbox, Inc., "Q4 2021 DBX Investor Presentation," Q4 2021. [Online]. Available: https://dropbox.gcs-web.com/static-files/b1e5d02b-6a9f-452d-be29-28a5632a3c2d.

[7] Dropbox, Inc., "Fourth Quarter 2021 Earnings Release," Q4 2021. [Online]. Available: https://dropbox.gcs-web.com/static-files/9ba48281-f330-451c-add8-5e28fded2ef6.
[8] T. Coughlin, "Digital Storage Projections for 2019, Part 3," 27 12 2018. [Online]. Available: https://www.forbes.com/sites/tomcoughlin/2018/12/27/digital-storage-projections-for-2019-part-3/.
[9] Dropbox, "Yesterday's Authentication Bug," 20 06 2011. [Online]. Available: http://web.archive.org/web/20110721173153/https://blog.dropbox.com/?p=821.
[10] BBC, "Dropbox hack 'affected 68 million users'," 31 08 2016. [Online]. Available: https://www.bbc.co.uk/news/technology-37232635.
[11] J. Raphael, "Google Docs Glitch Exposes Private Files," 09 03 2009. [Online]. Available: https://www.pcworld.com/article/527794/google_docs_glitch_exposes_private_files.html.
[12] S. Larson, "Data of almost 200 million voters leaked online by GOP analytics firm," 19 06 2017. [Online]. Available: https://money.cnn.com/2017/06/19/technology/voter-data-leaked-online-gop/index.html.
[13] R. Lakshmanan, "A Google Docs Bug Could Have Allowed Hackers See Your Private Documents," 29 12 2020. [Online]. Available: https://thehackernews.com/2020/12/a-google-docs-bug-could-have-allowed.html.
[14] Guardian News & Media Limited, "NSA Prism program taps in to user data of Apple, Google and others," 2013. [Online]. Available: https://www.theguardian.com/world/2013/jun/06/us-tech-giants-nsa-data.
[15] BBC, "Edward Snowden: Leaks that exposed US spy programme," 17 01 2014. [Online]. Available: https://www.bbc.co.uk/news/world-us-canada-23123964.
[16] The Internet Society, "PKCS #5: Password-Based Cryptography Specification Version 2.0," September 2000. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc2898.
[17] National Institute of Standards and Technology, "SP 800-57. Recommendation for Key Management: Part 1 – General," National Institute of Standards and Technology, Gaithersburg, MD, USA, 2020.
[18] M. B. Vaidya and S. Nehe, "Data security using data slicing over storage clouds," in 2015 International Conference on Information Processing (ICIP), Pune, 2015.
[19] P. Chiaro, S. Fischer-Hübner, T. Groß, S. Krenn, T. Lorünser and e. al., "Secure and Privacy-Friendly Storage and Data Processing in the Cloud," in Privacy and Identity Management. The Smart Revolution, M. Hansen, E. Kosta, I. Nai-Fovino and S. Fischer-Hübner, Eds., Ispra, Springer, Cham, 2018, pp. 153-169.
[20] R. Pottier and J.-M. Menaud, "Privacy-aware Data Storage in Cloud Computin," in 7th International Conference on Cloud Computing and Services Science (CLOSER 2017), 2017.
[21] P. Xu, X. Liu, Z. Sheng, X. Shan and X. Shan, "SSDS-MC: Slice-based Secure Data Storage in MultiCloud Environment," in 2015 11th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QSHINE), Taipei, 2015.
[22] Dr.K.Subramanian and F. John, "Secure and Reliable Unstructured Data Sharing in Multi-Cloud Storage using the Hybrid Crypto System," IJCSNS International Journal of Computer Science and Network Security, vol. 17, no. 6, pp. 196-206, 06 2017.
[23] K. Subramanian and F. L. John, "Dynamic and secure unstructured data sharing in multi-cloud storage," International Journal of Advanced and Applied Sciences, vol. 5, no. 1, pp. 15-23, 01 2018.
[24] F. Yahya, R. J. Walters and G. B. Wills, "Protecting Data in Personal Cloud Storage with Security Classifications," in 2015 Science and Information Conference (SAI), London, 2015.
[25] S. amamou, Z. trifa and M. khmakhem, "Data protection in cloud computing: A Survey of the State-of-Art," in Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 23rd International Conference KES2019, Budapest, 2019.
[26] P. Yang, N. Xiong and J. Ren, "Data Security and Privacy Protection for Cloud Storage: A Survey," IEEE Access, vol. 8, pp. 131723-131740, 07 2020.
[27] R. Wang, "Research on data security technology based on cloud storage," in 13th Global Congress on Manufacturing and Management, GCMM 2016, Hulunbuir, 2016.
[28] Alqahtani and D. H. Saad, A novel approach to providing secure data storage using multi cloud computing, Bedfordshire: University of Bedfordshire, 2019.
[29] Z. Huang, Q. Li, D. Zheng, K. Chen and X. Li, "YI Cloud : Improving user privacy with secret key," in 2011 IEEE 6th International Symposium on Service Oriented System (SOSE), Irvine, 2011.
[30] N. M. Joseph, E. Daniel and N. A. Vasanthi., "Survey on Privacy-Preserving Methods for Storage in Cloud Computing," in IJCA Amrita International Conference of Women in Computing, Amritanagar, 2013.

[31] O. Mir, R. Mayrhofer, M. Hölzl and T.-B. Nguyen, "Recovery of Encrypted Mobile Device Backups from Partially," in ARES 2018: Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg, 2018.
[32] J. Shen, X. Deng and Z. Xu, "Multi-security-level cloud storage system based on improved proxy re-encryption," EURASIP Journal on Wireless Communications and Networking, p. 277, 12 2019.
[33] M. Abdalla, M. Cornejo, A. Nitulescu and D. Pointcheval, "Robust Password-Protected Secret Sharing," in Proceedings of the 21st European Symposium on Research in Computer Security (ESORICS '16), Heraklion, 2016.
[34] M. Vaidehi and B. J. Rabi, "Design and analysis of AES-CBC mode for high security applications," in Second International Conference on Current Trends In Engineering and Technology - ICCTET 2014, Coimbatore, India, 2014.
[35] D. Jayasinghe, R. Ragel, J. A. Ambrose, A. Ignjatovic and S. Parameswaran, "Advanced modes in AES: Are they safe from power analysis based side channel attacks?," in 2014 IEEE 32nd International Conference on Computer Design (ICCD), Seoul, Korea (South), 2014.
[36] Microsoft Corporation, "Data Resiliency in Microsoft 365 - Microsoft Service Assurance | Microsoft Docs," 18 11 2021. [Online]. Available: https://docs.microsoft.com/en-us/compliance/assurance/assurance-data-resiliency-overview.
[37] Google LLC, "Disaster recovery planning guide," 24 08 2018. [Online]. Available: https://cloud.google.com/architecture/dr-scenarios-planning-guide.
[38] Dropbox, "Dropbox Business Security: A Dropbox Whitepaper," 2019. [Online]. Available: https://aem.dropbox.com/cms/content/dam/dropbox/www/en-us/business/solutions/solutions/white_paper/dfb_security_whitepaper.pdf.

AUTHORS

Artem Matveev, M.S. at the Siberian State University of Telecommunication and Information Science in Computer Science; Lead Engineer at the IT-Department and Teacher in Buryat Institute of Info communication.
His area of interest is wide and includes system and networking administration, software development, cyber security and electrical engineering.