Securing Cloud Data Under Key Exposure

[1]Mr.Akshay Agrawal, [2]Mr.Ketan Kadam, [3]Mr.Gyaneshwar Singh, [4]Mr.Jeetu Bhati

[1] Asst. Professor, UCoE, Kaman,[2,3,4]UG Student,[1,2,3,4]Computer Engg. Dept. Shivajirao S. Jondhle College of Engineering & Technology, Asangaon, Maharshatra, India.

[1]akshay1661@gmail.com, [2]ketank39@gmail.com, [3]gyaneshwarssingh@gmail.com, [4]jeetugamer1996@gmail.com

Abstract- As per the recent news it is disclosed that a hacker breaks data confidentiality by obtaining cryptographic keys, with the help of intimidation or backdoors in cryptographic softwares. If the encryption key is exposed, then the only workable measure to preserve data confidentiality is to limit the attacker's access to the ciphertext. This can be achieved, by randomly spreading the ciphertext blocks across servers in multiple administrative domains—thus preventing the adversary from compromising all of them. However, if existing schemes are used for data encryption, an adversary equipped with the encryption key, can still compromise a single server and decrypt the ciphertext blocks stored therein. In this proposed system, data confidentiality against an adversary is studied which knows the encryption key and has access to a large fraction of the ciphertext blocks. Bastion algorithm is proposed which is an innovative and efficient technique that guarantees the confidentiality of data even if the encryption key is compromised and the attacker has access to almost all ciphertext blocks. To study the security of Bastion and evaluate its performance by means of a prototype execution. Practical understandings with respect to the amalgamation of Bastion in commercial scattered storage systems is also addressed.

Keywords-Ciphertext, Securing, Cloud, Bastion, AONT.

I. INTRODUCTION

The world lately observed a massive surveillance program designed for breaking users' privacy. The culprits were not hindered by the various security measures installed within the targeted services. For instance, even though these services are dependent on encryption mechanisms for assuring data confidentiality, the essential keying material is attained by the means of backdoors, bribe, or coercion. If the encryption key is compromised, the only practical mean to assure confidentiality is to limit the adversary's access to the cipher-text, e.g., by scattering it across multiple administrative domains, in the hope that the adversary cannot access all of them.

Even after the encryption of data and scattering it across multiple administrative domains, an attacker armed with the suitable keying tool can hijack a server in one domain and decrypt the cipher-text blocks stored in it. In proposed system, data confidentiality will be protected against an adversary who knows the encryption key and has access to a big fraction of the cipher-text blocks. [2]

To prevent such an adversary, Bastion is recommended, it is an innovative and effective technology which ensures that plaintext data cannot be retrieved until the adversary has the access to at most all but two cipher-text blocks, even when the encryption key is compromised.

II. AIMS AND OBJECTIVE

a) Aim

The reason why the aim is concentrating on cloud computing is that lately the use of cloud from running a service to storing data has been increasing day by day. Hence there needs to be more concentration on cloud and also the necessity for security in cloud is more. This is because a lot of cloud service providers' offer free memberships and because of this the security aspect is being compromised. So from the user point of view it is essential that some security features are introduced to safeguard users' data.

b) Objective

- **Helping users to maintain their data private/safe:** This application will provide privacy to users' data by means of encryption.
- **To reduce liabilities in cloud computing:** This application will help reduce a number of liabilities in cloud computing.
- **Provide security against attackers/hackers:**

  This application will provide much better security against intruders, extruders, and all kind of attackers who are willing to access data without permission.

 DOI : 10.35291/2454-9150.2020.0223

III. LITERATURE SURVEY

1) "EFFICIENTLY UTILIZING THE ERASURE CODES IN A DISPERSED SYSTEM"

Erasure codes offer space-optimal data redundancy for protection against data loss. It is generally used in a dispersed system for the dependable data storage, in which the data that are erasure-coded are stored in various nodes to stand node failures without any loss of data. This paper recommending a new method to preserve erasure-encoded data in a distributed system. This method permits the use of space efficient k-of-n erasure codes where n and k are huge and the overhead n-k is minimal.[1]

2) "SOLIDIFYING SECURITY BY COMPOSITION: THE CASE OF MAGNIFIED IDEAL CIPHERS"

In the Shannon model; the safety of constructions equivalent to double and (two-key) triple DES. That is, consider Fk1 (Fk2 ()) and Fk1 (F 1k2 (Fk1 ())) with the component functions being perfect ciphers. This model the defiance of these constructions to common attacks like man in the middle attack. Obtained the first proof that composition actually increases the security in some meaningful sense.[2]

3) "ROBUST DATA SHARING WITH KEY-VALUE STORES"

A key-value store (KVS) provides functions for storing and recovering values associated with unique keys. KVSs have become the trendiest way to access Internet-scale "cloud" storage systems.

Providing a competent and innovative algorithm that matches multi-reader multi-writer storage from a group of presumably faulty KVS copies in an asynchronous setting. The implementation of this technology can help in serving a limitless number of customers parallelly using the storage. It stomachs crashes of a marginal of the KVSs and crashes of any number of users.

Compared to previous solutions, it is inherently scalable and allows clients to write alongside. Because of the restricted interface of a KVS, textbook-style results for dependable storage either don't work or incur too massive overhead. Storage algorithm maintains two copies of the stored value per KVS in the common case, and we show that this is indeed necessary.[3]

IV. EXISTING SYSTEM

If the encryption key gets exposed, the sole viable means to ensure confidentiality is to limit the adversary's access to the cipher-text, e.g., by spreading it across multiple administrative domains, within the hope that the adversary cannot compromise all of them. However, albeit the information is encrypted and dispersed across different administrative domains, an adversary equipped with the relevant keying material can compromise a server in one domain and decrypt cipher-text blocks stored therein.

Ramp schemes constitute a trade-off between the safety guarantees of secret sharing and therefore the efficiency of data dispersal algorithms.

A ramp arrangement achieves more "code rates" compared to secret sharing and features two thresholds t1, t2. A minimum of t2 shares are required to reconstruct the key and fewer than t1 shares provide no information about the secret; variety of shares between t1 and t2 leak "some" information. Combined AONT and data dispersal to give both fault-tolerance and data secrecy, within the context of distributed storage systems.

In existing system, however, an adversary which knows the encryption key can decrypt data stored on single servers [4]

V. COMPARTIVE STUDY

Table number:1 Comparative Analysis of Existing Systems

Sr. No	Paper Name	Author/ Publication	Technology	Advantage	Disadvantage
1.	Securing Cloud Data Under Key Exposure.[4]	Ghassan O. Karame, Claudio Soriente,	Bastion algorithm	Comparatively less over-head (<5%) as compared to other existing systems.	Cannot access applications running on a private network.
2	Using Erasure Codes Efficiently For Storage.[1]	M. K. Aguilera, R. Janakiraman, and L. Xu.	Space efficient k-of-n Method	The use of space efficient k-of-n erasure codes.	No two-phase commits and no logs of old versions of data.
3.	Security Amplification by Composition.[2]	W. Aiello, M. Bellare, G. D. Crescenzo, and R. Venkatesan.	Shannon model	Increased security against attacks like in the middle attacks.	Doesn't provide security against all attacks.
4.	Robust Data Sharing With Key-Value Stores.[3]	C. Basescu, C. Cachin, I. Eyal, R. Haas, and M. Vukolic.	Key-value store (KVS)	Algorithm minimizes the space overhead at the KVSs.	Maximum space complexity grows with concurrent write operations.

 DOI : 10.35291/2454-9150.2020.0223

VI. PROBLEM STATEMENT

The security issues in cloud computing includes:

- Data security
- Identity and access control
- Key management
- Virtual machine security

Among these main security issues in the cloud, data security and integrity are believed to be the most difficult problem which could limit the use of cloud computing. The access control and key management are two of the most important issues involved in data security. Data security in the cloud refers to data confidentiality, integrity, availability and traceability (CIAT), and these requirements pose major problems for cloud computing.

VII. PROPOSED SYSTEM

In proposed system, information classification against an enemy which knows the encryption key and approaches a huge part of the cipher-text blocks is examined.

To this end, Bastion will be used, it is a novel and effective plan that ensures information privacy regardless of whether the encryption key is spilled and the enemy approaches all cipher-text blocks. In this proposed system security of Bastion will be dissected and it will be assessed by execution by methods for model usage. Additionally, examination of functional bits of knowledge regarding the mix of Bastion in business scattered capacity systems will be done.

Conducted assessment results propose that Bastion is appropriate for joining in existing systems since it costs overhead fewer than 5% contrasted with existing semantically secure encryption modes.

In proposed system, data confidentiality will be preserved against an adversary who knows the encryption key and has access to a large fraction of the cipher-text blocks.

The adversary can acquire the key either by exploiting flaws or backdoors in the key-generation software, or by compromising the devices that store the keys.

To counter such an adversary, Bastion is proposed, it is a novel and efficient scheme which ensures that plaintext data cannot be recovered as long as the adversary has access to at most all but two cipher-text blocks, even when the encryption key is exposed.[4]

VIII. ALGORITHM

A. Encryption in Bastion:

1: procedure Enc(K, x = x[1] . . . x[m])
2: n = m + 1
3: y ′ [n] ← {0, 1} l ⊲ y'[n] is the IV for CTR
4: for i = 1 . . . n − 1 do
5: y ′ [i] = x[i] ⊕ FK(y ′ [n] + i)
6: end for
7: t = 0l
8: for i = 1 . . . n do
9: t = t ⊕ y ′ [i]
10: end for
11: for i = 1 . . . n do
12: y[i] = y ′ [i] ⊕ t
13: end for
14: return y ⊲ y = y[1] . . . y[n]
15: end procedure

B. Decryption in Bastion:

1: procedure Dec(K, y = y[1] . . . y[n])
2: t = 0l
3: for i = 1 . . . n do
4: t = t ⊕ y[i]
5: end for
6: for i = 1 . . . n do
7: y ′ [i] = y[i] ⊕ t
8: end for
9: for i = 1 . . . n − 1 do
10: x[i] = y ′ [i] ⊕ F −1 K (y ′ [n] + i)
11: end for
12: return x ⊲ x = x[1] . . . x[n − 1]
13: end procedure

IX. MATHEMATICAL MODEL

An All or Nothing Transform (AONT) is an efficiently computable transform that maps sequences of input blocks to sequences of output blocks with the following properties:

(i) Given all output blocks, the transform can be efficiently inverted.

(ii) Given all but one of the output blocks, computing any of the original input blocks is not feasible.

A pair of p.p.t. algorithms gives the formal syntax of an AONT Q= (E, D) where:

E: The encoding algorithm is a probabilistic algorithm which takes as input a message x ∈ {0, 1} ∗, and outputs a pseudo-cipher-text y.

D: The decoding algorithm is a deterministic algorithm which takes as input a pseudo-cipher-text y, and outputs either a message x ∈ {0,1}∗ or ⊥ to indicate that the input pseudo-cipher-text is invalid.

For correctness, all x ∈ {0, 1} ∗, and for all y ← E(x),

Given: x ← D(y).

$Exp_{\pi}^{aont}(A, b)$

x,state← $A(find)$

$y_0 \leftarrow E(x)$

$y_1 \leftarrow \{0,1\}^{|y_0|}$

b' ← A^{Y_b}

 DOI : 10.35291/2454-9150.2020.0223

X. SYSTEM ARCHITECTURE

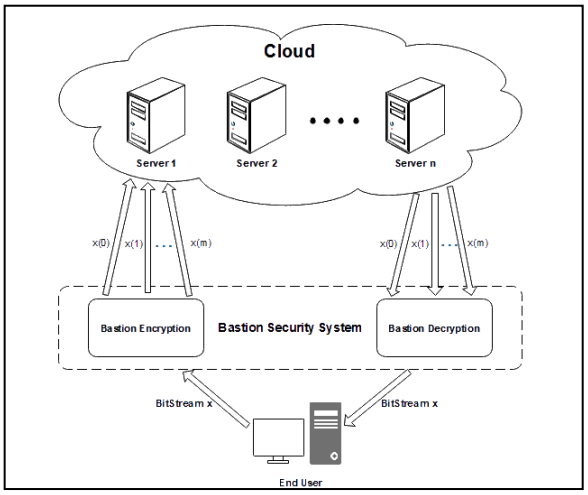

Fig.1: System Architecture

Description: In proposed system, data confidentiality will be preserved against an adversary who knows the encryption key and has access to a large fraction of the cipher-text blocks. The adversary can acquire the key either by exploiting flaws or backdoors in the key-generation software, or by compromising the devices that store the keys. Bastion security is a novel and efficient scheme which ensures that plaintext data cannot be recovered as long as the adversary has access to at most all but two cipher-text blocks, even when the encryption key is exposed. The output blocks have a secret key embedded in it which is leveraged by the majority of AONTs. Once all output blocks are available, the key can be recovered and single blocks can be inverted.[4]

XI. ADVANATGES

- The performance of Bastion is evaluated in comparison with a number of existing encryption schemes. The results show that Bastion only incurs a negligible performance deterioration (less than 5%) when compared to symmetric encryption schemes, and considerably improves the performance of existing AON encryption schemes.
- Bastion ensures data confidentiality against an adversary that knows the encryption key and has access to a large fraction of the cipher-text blocks.
- Preventing leakage of any plaintext block as long as the adversary has access to the encryption key and to all but two cipher-text blocks.
- Bastion considerably improves the performance of existing AON encryption schemes, and only incurs a negligible overhead when compared to existing semantically secure encryption modes (e.g., the CTR encryption mode).
- Practical insights with respect to the deployment of Bastion within existing storage systems, such as the Hydrator grid storage system has been addressed.
- The performance of Bastion analytically and empirically in comparison to a number of existing encryption techniques is evaluated.

XII. DESIGN DETAILS

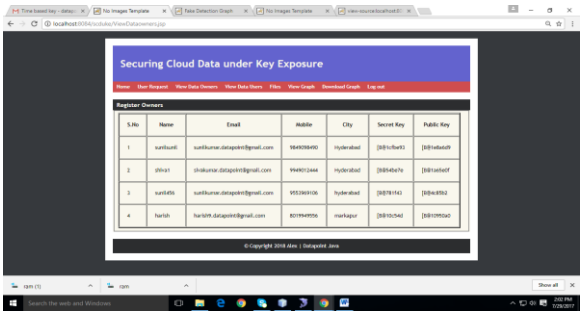

Fig.2: Data owners

XIII. CONCLUSION

Thus, we have tried to implement "Securing Cloud Data Under Key Exposure" by Ghassan O. Karame, Claudio Soriente, Krzysztof Lichota, Srdjan Capkun. A novel security definition is used that captures data confidentiality against the new adversary. Bastion is introduced, which is a scheme that ensures the confidentiality of encrypted data even when the adversary has the encryption key, and all but two ciphertext blocks. Bastion is most suitable for settings where the ciphertext blocks are stored in multi-cloud storage systems. Bastion considerably improves the performance of existing primitives which offer comparable security under key exposure.

REFERENCE

[1] M. K. Aguilera, R. Janakiraman, and L. Xu. "Encryption for Algorithm for Cloud Computing" 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference IEEE, 2008.

[2] L. Lamport, "On interprocess communication," 1985.

[3] V. Boyko, "On the Security Properties of OAEP as an Allor-nothing Transform," in Advances in Cryptology (CRYPTO), 1999, pp. 503–518.

[4] G. R. Blakley and C. Meadows, "Security of ramp schemes," in Advances in Cryptology (CRYPTO), 1984, pp. 242–268. [5] A. Shamir, "How to Share a Secret?" in Communications of the ACM, 1979, pp. 612–613.

[6] NEC Corp., "HYDRAstor Grid Storage," http://www.hydrastor.com.

[7] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, "Deniable Encryption," in Proceedings of CRYPTO, 1997.

[8] J. H. van Lint, Introduction to Coding Theory. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1982.

 DOI : 10.35291/2454-9150.2020.0223