ISSN22773061

A NOVEL TECHNIQUE FOR TRUST DELIVERY IN THE CLOUD

Y Lakshmi Prasanna[1], Dr.E.Madhusudhana Reddy[2], S Neelima[3]
[1] Department of Computer Science and Engineering, Research Scholar, JNTUH, Hyderabad, India.
prasanna.yeluri@gmail.com
[2]Department of Computer Science and Engineering, Madanapalle Institute of Technology and Science, Madanapalle - 517325, India.
e_mreddy@yahoo.co.in
[3]Department of Computer Science and Engineering, Research Scholar, JNTUH, Hyderabad, India.
sarabu.neelima@gmail.com

ABSTRACT:

For many organizations, keeping data private and secure has also become a compliance requirement. Cloud providers offering Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) offer a “shared responsibility” model for customer applications and data, so companies that are migrating to the cloud are responsible for finding a solution. This research paper proposes a system, which combines encryption with key management to protect critical data in public, private and hybrid cloud environments.

Keywords: Cloud Compliance, Cloud Security, Data Encryption, Key management, Threats

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTER AND TECHNOLOGY

Vol 11, No. 2

editor@cirworld.com

www.cirworld.com, member.cirworld.com

1. INTRODUCTION:

There are many compelling reasons to migrate applications and data to private or public clouds: scalability, agility, cost savings etc. Any organization that is migrating data to the cloud needs to manage the risk to data at rest with a robust solution for data encryption and encryption-key management. Securing data – at rest and in use – is simpler when it is located within the four walls of a data center. Once it is moved to the cloud, it becomes vulnerable to a number of new threats ranging from stolen administrator credentials to new hacking techniques. For many organizations, keeping data private and secure has also become a compliance requirement. Cloud providers offering Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) offer a shared responsibility model for customer applications and data, so companies that are migrating to the cloud are responsible for finding a solution. In this context introducing the risks to data in the cloud comes up with an idea of proposing a system, which combines encryption with key management to protect critical data in public, private and hybrid cloud environments. This system mainly uses three core technologies to deliver trust in the cloud:

- Robust, standards-based data encryption with a convenient, fast and simple management interface.

- Cloud-ready key management using Split-Key Encryption

- Homomorphic key encryption techniques that protect keys even when they are in use.

Each of the above plays a vital role in ensuring that the data is safe and encryption keys are protected, both when in storage and when in use in the cloud. Together, the proposed system can be treated as a solution that offers the convenience of encryption and key management in virtualized environments.

2. CHALLENGES OF SECURING DATA IN THE CLOUD

Data encryption is one of the most important methods of protecting data at rest in the cloud. In order to select the most effective solution, it's important to understand the primary challenges of encrypting data in the cloud.

2.1 Managing the encryption process

For complex applications with large amounts of data, the most time-consuming aspect of data encryption is management: deployment, set-up, adding and removing disks, etc. An effective encryption solution can reduce the time required from each of these tasks from hours to minutes.

2.2 Securing the data lifecycle

Whether in use or at rest, both the data and the encryption keys must be handled and deployed correctly. An effective encryption solution must address every stage in the lifecycle.

2.3 Delivering high performance

To ensure that the quality of service for the cloud applications meets expectations, the encryption solution must offer very high performance.

2.4 Ensuring trust in the cloud

The problem with hosting key management in the cloud is one of trust. For both security and compliance reasons, one cannot afford to allow a third party to manage your keys. In order to benefit from the convenience and low cost of cloud-based key management, a sophisticated solution that leaves the root of trust in hands alone is needed.

2.5 Storing and managing the encryption keys

Every time the application accesses the data store, it needs to use encryption keys. There is generally one key per disk or data store and all of them must be managed on a key management server. Hosting a key management server in the data center is expensive, undermining the cost benefits of cloud applications.

2.6 Protecting the keys from theft when they are in use

Encryption keys are vulnerable at two points – when in storage, and while in use. A truly effective key management solution will be able to protect the keys at both times.

3. DATA ENCRYPTION

The most secure data encryption solutions must support all of the major business systems are full disk encryption, database encryption, file system encryption, distributed storage encryption and even row or column encryption. The system above applies the same encryption technology to all of these needs.

Whenever an application (such as a database server) writes a disk block, it goes through the virtual data system, where the data is encrypted and sent to the disk volume. The plain text data is never written to persistent storage. All requests to read data from the disk get sent to the virtual data system, which reads the encrypted data blocks, decrypts them and then sends the plain text data back to the requesting application.

The system provides the unique ability to invisibly hook the encryption solution between the data storage and the application or database servers in the cloud. Once permission is granted, the encryption solution is transparent to the application and can be integrated quickly and easily without any application changes at all. The system uses Advanced Encryption Standard (AES) encryption algorithm with a 256-bit key. Multiple blocks are chained using Cipher-Block Chaining (CBC), and the Encrypted Salt-Sector Initialization Vector (ESSIV) scheme is used to counter so-called fingerprinting attacks. It is also possible to configure the system to use alternate encryption algorithms as needed.

The Proposed solution can also encrypt several different types of data on-the-fly:

- Disk volumes, which can be exposed to applications as Network File System (NFS) disks, or as Windows shares (CIFS volumes).
- Disk volumes configured as a Storage Area Networks (SAN). This is a common way to configure storage for database servers.
- Distributed storage, where applications normally write the whole file into a Web Service, and benefit from extremely high durability.

4. KEY MANAGEMENT USING SPLIT - KEY ENCRYPTION

As the entire operation is dependent upon the security of the keys, it is sometimes appropriate to devise a fairly complex mechanism to manage them. If a single individual is involved, often direct input of a value or string is sufficient. The memorized value will then be provided as re-input to retrieve the data, similar to password usage. Sometimes, many individuals are involved, with a requirement for unique keys to be sent to each for retrieval/decryption of transmitted data. In this case, the keys themselves may be encrypted. To encrypt data, the system performs the encryption algorithm on both the plain text and the secret key to obtain the cipher text: C = EK(P). The best practice is to generate as many different random keys as practical - e.g. one key per disk volume or object - and to store them securely. Storing the key next to the encrypted data would be vulnerable to the same attacks as the data. But with the case of cloud applications or systems, one does not want to store the keys in the cloud with the data, but of course they are needed to access data stored on the application servers and database servers. Here the cloud based hosted key management system has to handle this situation without sacrificing trust. This is achieved by using Split-Key Encryption Technique

Split-Key encryption technique protects the keys and guarantees that they remain under customer control and never exposed in storage. The split-key encryption is similar to the traditional practice used to protect private safe-deposit boxes at banks around the world. Each safe-deposit box has two keys: one is held by the customer, the other is kept by the bank. Neither the customer nor the banker can open the safe on their own; both keys are needed at the same time.This Key Management solution requires two keys. Each data object (such as a disk or file) is encrypted with a unique key that is split in two. The first part – the Master Key – is common to all data objects in the application. It remains the sole possession of the application owner and is unknown to the Data System. The second part is different for each data object and is stored by the Key Management Service provided by the system. When the application accesses the data store, the system uses both parts of the key to dynamically encrypt and decrypt the data.

Whenever a new application is developed a single Master Key is generated and securely it should be kept as a back up. The Master Key is used by the system which resides in own cloud account, but it is never transferred into the system's Key Management Service. When encryption of a disk volume is done, it receives a new key that is a mathematical combination of the Master Key and a unique random key created by the system and stored in an encrypted form in the Key Management Service block. So for each application, the user has to keep track of one master key. For every disk or data storage object used by the application, the system takes care of generating the second half of the key, and stores it in the Key Management Service block after further encryption with a private key. To retrieve the encrypted key the system combines the Master Key with the Second Key to obtain a key that will actually decrypt an object. When an ongoing access to a data object is no longer required the Management Interface (or API) can be used to lock the object. The key is then erased, and only the encrypted part is retained in Key Management Service. The object is still protected by both the Master Key and the Encrypted Key. When the key is needed again, it will be fetched from the key management service block.

5. PROTECTING ENCRYPTION KEYS USING HOMOMORPHIC ENCRYPTION

Virtual Key Management is the only solution that keeps data and encryption keys safe at all times – even when they are in use in the cloud. Homomorphic key encryption is a technique that enables mathematical operations to be performed on the encrypted data. This Key Management enables the system to give the application access to the data store without ever exposing the master keys in an unencrypted state.

As explained above, each data object is encrypted with a key that has two parts: the Master Key and the Second Key. When the application needs to access the data store, the System combines both parts of the key by using a mathematical operation. This would require both parts of the keys to be unencrypted and exposed. However with this system, both parts of the key are encrypted before and during their use in the system. As a result, the keys are fully encrypted when they are resident in a user's cloud account. This technique homomorphically encrypts the master key differently for each instance of the system. So even if the user's cloud account is breached or attacked, and the encrypted master key is stolen, it can never be used to access the data. With Fully Homomorphic Encryption, all mathematical operations can be performed on encrypted data, but since it requires an enormous amount of computational resources, it is not so feasible. With Partially

Homomorphic Encryption, only selected mathematical operations are supported, dramatically reducing the computational overhead. The Proposed solution consists of Partially Homomorphic Encryption so that the most critical link in the encryption of data in the cloud – the master key – is also encrypted and secure. At the same time, users of the system are benefitted from fast, reliable performance for their business-critical applications.

6. THREATS IN THE CLOUD

Threats to Cloud Security are widely publicized and they are real; but with this proposed solution, a level of data protection that is unavailable even in on-premise encryption solutions can be obtained.

All data encryption systems, both in the cloud or in a physical data center, share a common vulnerability – they need to use the encryption keys and when the keys are in use, they can, in theory, be stolen.

Generally Cloud applications are designed for security. The disks never contain the encryption keys and the memory is inaccessible – even to the owner. Nevertheless, in the highly unlikely event that a Cloud System is breached and the encryption key is stolen, only the one data object that is in memory at that time is exposed. In order to access the rest of user's data storage, the thief would need the Master Key.

7. CONCLUSION

Data encryption is crucial in providing security to the data in the cloud. But encrypting the data is only the beginning, where as managing and protecting the encryption keys effectively is vital. An effective data encryption solution must include:

- Robust, fast, yet easy to use data encryption.
- Reliable, cloud-based key management that is cost-effective, but trustworthy.
- Key encryption technologies to protect encryption keys as well as data – both in storage and in use.

The proposed work is the solution to offer cloud-based key management without sacrificing trust. The system requires two parts of a split key to access every disk. Each part of the key is encrypted to protect it while it is resident in the user's cloud account using patent-pending homomorphic key encryption technology. With the Master Key, user can retain control of the encrypted data, without having to install and maintain key management servers on premise.

ACKNOWLEDGMENT

The authors would like to acknowledge and express their heartfelt gratitude to the Swarna Bharathi Institute of Science and Technology [SBIT] Management, Chairman, Director, Principal and Dean R&D. for their vital encouragement, support and facilities providing by them to achieve this task and at they would like to thank Anonymous Reviewers for their valuable suggestions and comments. This paper has greatly benefited from their Efforts.

REFERENCES

[1]. S.Neelima, Y.Lakshmi Prasanna and M.Padmavathi, Scenarios for Leveraging Predicate Based Encryption (PBE) in the Cloud, International Journal For Computer Science and Technology (IJCST), September 2012, Vol. No. 3, Issue 3 pp 180-185.

[2]. Key Management: A Cryptography Tutorial, http://www.cryptographyworld.com/key.htm

[3]. Hsiao-Ying Lin, Wen-Guey Tzeng, National Chiao Tung University, Hsinchu City, A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding, IEEE Transactions on Parallel and Distributed Systems June 2012, (vol. 23 no. 6) pp. 995-1003.

[4]. Cloud data storage: New technologies make cloud storage more appealing, TechTarget.

[5]. Storing encryption keys — best practices, http://stackoverflow.com//storing-encryption-keys-best-practices

[6]. Katti, R.S., North Dakota State Univ., Fargo, ND, USA ,On the Security of Key-Based Interval Splitting Arithmetic Coding With Respect to Message Indistinguishability, IEEE Transactions on Information Forensics and Security Volume: 7, Issue: 3,ISSN :1556-6013

[7]. Fakhar.F , Shibli.M.A, School of Electrical Engineering & Computer Science, National University of Science & Technology, Islamabad, Pakistan, Management of Symmetric Cryptographic Keys in Cloud Based Environment, Advanced Communication Technology (ICACT), 2013 15th International Conference, ISSN:1738-9445.

[8]. Disk Encryption Theory, http:// en.wikipedia.org/ wiki/Disk_encryption_theory

[9]. Jesse Kornblum, Practical Methods for Dealing with Full Disk Encryption

[10]. Dawn Xiaodong Song, David Wagner, Adrian Perrig, Practical Techniques for Searches on Encrypted Data.

Mrs. Y. Lakshmi Prasanna, working as an Associate professor in the Department of Information Technology, pursuing her Ph.D. in Computer Science and Engineering from JNTUH, Hyderabad. Her research areas include Cloud Computing, Computer Networks, Mobile Computing and Data Warehousing and data mining.

Dr.E.Madhusudhana Reddy working as a Professor in the Department of Computer Science & Engineering. His areas of specialization includes Cryptography & Network Security, Biometrics, Data Mining and Warehousing, Artificial Intelligence & Neural Networks and Human Computer Interaction

Mrs. S. Neelima, working as an Associate professor in the Department of Information Technology, pursuing her Ph.D. in Computer Science and Engineering from JNTUH, Hyderabad. Her research areas include Cloud Computing, Data Ware housing and data mining, Computer Networks and Network Security.