International Journal of Engineering Innovation & Research
Volume 8, Issue 1, ISSN: 2277 – 5668

Secured Key Menagement in Manufacturing Execution Systems of Cloud Connect

Renu[1*], Dr. Sanjeev Sharma[2] and Dr. Vishnu Sharma[3]
[1]School of Information Technology Rajiv Gandhi Technological University, Bhopal, India ;
*Corresponding author email id: renutrivedi@rediffmail.com
[2]School of Information Technology Rajiv Gandhi Technological University, Bhopal, India ;
email id: sanjeev @rgtu.net
[3]Department of computer science Galgotia College of Engineering, Greater Noida, India ;
email id: vishnu..sharma @galgotiacollege.edu

Date of publication (dd/mm/yyyy): 15/01/2019

***Abstract* – In manufacturing organizations, the flow of information often is essential to the functioning of critical design, processes, machinery, or systems on which the company depends. Thus, Cyber security rather than being an option becomes a necessity as critical piece of every system in the organization. For many manufacturers, integrating plant floor data with enterprise systems poses sufficient challenges. It's no wonder that one of the biggest reason manufacturers are hesitant to connect plant floor systems to the corporate network and extend Internet connectivity to production data is security. The data sharing is one of the concerning functionality in today's world as it involves security, efficiency and flexibility as their important aspects. With the introduction of encryption and decryption schemes, the storing, sharing and securing of data became rampant. The storing of these cipher texts and the decryption keys is one of the major issues. There is a need for a mechanism which can minimize the cost of storing these cipher texts and keys in a secured way. Protecting user's data privacy is one of the critical goals of cloud storage. This research effort focuses more on aggregation of these keys into a single aggregate key which will in turn reduce the burden on the network overhead.**

***Keywords* – Data over Cloud, Public-Key Encryption, KAM technique.**

I. INTRODUCTION

The cloud offers particular benefits for industrial enterprises that depend on remote monitoring of devices. Sometimes the company wants the capability of remotely monitoring its plants, which often run in remote areas where there are no IT servers or Internet connectivity, using a connect way into the cloud via cellular or satellite connections. With this cloud-based solution, the company has a highly scalable, cost-efficient method to store and remotely access real time information to help extend equipment lifecycles and provide improved value to customers. In today's technological environment, as companies become more connected both internally and externally, the possibility of security problems increases, requiring an industrial security strategy that is deeply entrenched within both the plant and the broader enterprise. Our discussion in this paper focuses on the problem statement and exploration of the approaches. Specifically, we will discuss potential vulnerabilities to special type of attacks during task dispatching, and design defense mechanisms to balance user secrecy with system efficiency. Through exploring security vulnerabilities caused by the sharing and designing defense mechanisms, we expose insights of cloud security and provide guidelines with defense mechanisms. The lack of research efforts in system security and robustness, however, poses a serious challenge to future adoption of cloud manufacturing. Our proposed research tries to enhance cloud manufacturing security through key management approaches during the task planning and production phases. The present research efforts concentrates more on aggregation of encryption keys into a single aggregate key which will in turn reduce the burden on the network overhead. In this paper, we'll explore what the landscape of cloud encryption key management looks like today.

In Section 2, we tried to cover the work already done in this field later on we will introduce the concept of cloud manufacturing and the research challenges in security as section 3. In Section 4, we investigate the protection of user privacy from different attacks. Section 4 describes the development of a secured platform. At last, Section 5 concludes the paper.

II. LITERATURE SURVEY

In previous work [3], an effective TPA (Third Party) is introduced so that it would not bring any probability of attack on user data privacy, and will also not put an additional burden to user. In cloud storage, users can store their data and utilize the various resources, services or applications without causing burden of local data storage and its maintenance. However, it becomes a problem for users with limited computing resources. These users must not worry about the need to verify data integrity and use the cloud storage. This made public auditability for cloud storage very important so that users can employ a Third Party Auditor (TPA) for checking the integrity of outsourced data. High security and performance analysis show that this scheme is secure and highly efficient. The storage correctness and privacy-preserving features were given higher importance. Whenever the user is not completely happy with trusting the security or honesty of technical staff, they are motivated to encrypt their data with their own keys before dumping them to the server. In 2013 Jens-Matthias Bohli, Nils Gruschka, Meiko Jensen, Luigi Lo Lacono and Ninja Marnau, “Security and Privacy-Enhancing Multicloud Architectures”, attacks on data and applications in a public cloud is the simultaneous usage of multiple clouds. Several approaches employing this paradigm have been proposed recently this paper provides a survey on different security by multicloud adoption approaches. It provides four different models in form of abstracted multicloud architectures. These developed multicloud architectures allow to categorize the available schemes and to analyze them according to their security benefits. [12] in [5], Yogita Gunjal et al. have presented new flexible and effective scheme for data integrity in terms of correctness for distributed storage system which removes or corrects code in the file distribution preparation to support redundancy parity vectors for verification of removed coded data using the homomorphic token .in cloud which consists of correctness insurance for storage integration, localization for data errors, data block dynamic operation. It gives efficient outputs against alternation attack and Byzantine failure. In [6], Gangolu Sreedevi et al. have presented new way of security ICCC for small organizations in cloud where data storage transparency can be minimized. They are not encrypting the whole message. Rather than encrypting whole message, they encrypt the some bits in each block. For data correctness, they have generated the Meta data which is used to verify the data for alteration or deletion by unauthenticated party. Through this technique, they achieve the correctness of data of owner at low cost and computational part is also free from overhead. In [7], Rupali Sachin Vairagade et al. have focus on the security of data in the cloud. Thus they have derived main three areas location of data, control of data and secure transfer of data in cloud, where data security is more concern. For security of the data, they have presented conversion of plaintext to ASCII Code and then the public key encryption RSA algorithm of ASCII code instead of plain text. So, only numeric data is to be encrypted instead of characters. In [8], Kalpana Batra et al. have tried to achieve the security of data in distributed storage system by applying the file distribution technique to provide the redundancy. For correctness of the data they have generated the token pre computation technic and stored at servers in cloud for the verification purpose. They have shown that their scheme is efficient and reliable to detect the misbehaving servers and correct the data in particular servers and avoid colluding attacks of server

modification by unauthorized users. In [9] K.RAJASEKAR et al. have proposed a secured cost-effective multi-cloud storage (SCMCS) model which consist of multiple cloud which provides the high availability and security compare to single cloud with economical cost to customer. Data is available among multiple SPs as per the budget of the customer. To ensure data availability, the user's data block is divided into various data pieces and distributed among the available SPs in such a way that no SP can make successful retrieval which have meaningful information. In [10], Qian Wang et al. have focus on the problem of simultaneous public audit of data and data dynamics on cloud storage for the integrity of data. They provide the solution which provides the efficiency for data dynamics by improving the storage models by manipulating the classic Merkle Hash Tree construction. They have used bilinear aggregate signature for the achieving their main goal of simultaneous multiple auditing task. In [4], Rakhi Bhardwaj et al. have represented dynamic data auditing policy for the data storage on the cloud. They have discussed on auditing model for the data storage in cloud which consists of data owner auditing, Third Party Auditing (TPA) and derived the resulting research where TPA can support the dynamic auditing for multiple tasks simultaneously. Thus, the high performance on data availability and integrity can be achieved.

III. ADVANTAGE AND CHALLENGES OF CLOUD MANUFACTURING

An advantage of cloud manufacturing is the sharing of resources. Such sharing can highly improve the equipment usage efficiency and allow middle or small scale manufacturers to conduct tasks that used to be impossible. The sharing of resources, however, also creates a channel through which malicious attackers can steal information from other users or gain advantage during the competition.

Although some research efforts [33] treat cloud manufacturing as a natural extension of cloud computing, but we know that the resources in cloud manufacturing have at least the following differences from those in cloud computing.

- First, while virtual machine migration in cloud computing is almost free [34], shipping of parts in cloud manufacturing causes extra delay and costs. Therefore, we have to consider the physical distances among the resource providers that we choose.
- Second, although in cloud computing you can interrupt and recover a virtual machine almost in stantly with very little performance penalty, in cloud manufacturing it is usually very costly to interrupt a task.
- Last but not least, while the capabilities of the physical boxes in a computing cloud are almost the same (either CPU cycles or storage spaces), the capabilities of the manufacturing equipments vary greatly.

We investigate this problem to identify the potential threats and mitigate their impacts. The main drawback, however, is security and in particular data confidentiality. Users of cloud technology essentially have to trust that the cloud providers do not misuse their data. In the simple cloud computing case where a user outside the cloud wants to store some data in the cloud for later retrieval, data confidentiality and integrity can relatively easy be ensured. This is typically done using standard cryptography, by encrypting the user's data before it is stored in the cloud, keeping the encryption key secret from the cloud provider.

IV. CLOUD KEY MANAGEMENT

Today, the challenges of cloud encryption key management are still a major barrier to storing sensitive data within cloud provider environments. Cloud providers and consumers are starting to solve this problem, however, and it is likely that key management will be a major focus area for cloud security in the coming months and years. The primary difference between key management in an enterprise's data center versus key management in the cloud is ownership and management of the keys. In a traditional data center, all key management functions and tools can be configured and maintained by an IT operations team. In cloud environments, there will likely be a shared model or one wholly managed and maintained by the providers.

4.1. *Issues in Cloud Key Management*

Cloud key management processes will largely depend on several factors. In some cases, the type of cloud service in use will dictate the types of key management available. IaaS clouds have internal key management maintained for digitally signing virtual machine image templates. Public key infrastructure (PKI) is used for signing API commands and for gaining access to VM images. The private keys in this arrangement need to be maintained by the cloud consumer and can be stored internally within traditional key management platforms. For PaaS and SaaS clouds, most key management functions are managed internally at the cloud provider, though private keys for access to applications and systems can be distributed to consumers for data, application or database access to cloud resources. In public key deployments, the key management and security is maintained by private keys distributed to consumers are controlled by the consumers. Any other internal key management will largely be the provider's responsibility. For hybrid clouds, key management is most likely shared, and private clouds typically have key management tools and processes within the internal network environment. As the sensitivity of data moving into the cloud increases, security professionals are actively looking to protect this data using encryption, with tried-and-true techniques they've been using in their data centers for years. In some cases, however, this may not be possible or may require some different approaches and tools, especially for encryption key management.

4.2 *Need for Key Aggregation:*

Key aggregation plays an important role in handling the overhead on networks. With the increase in usage of different devices and systems, the traffic on networks is increasing. For an example, an employee keeps her private data i.e. designs on Dropbox and he doesn’t want to share it with everyone. Due to the possibility of data leakage, he can not only depend on the privacy preserving mechanism provided by Dropbox, so before uploading the designs he encrypts them by his own key. Some day he wants to share designs to another person X in which X is interested. He can share the designs by using the share function of the Dropbox, but the problem is how to share the decryption rights. One option is he sends an email in which the private key is involved. There are 2 possibilities: [5]

1. Designer can encrypt all designs with one key and send securely.
2. Designer encrypts each photo with different keys and sends the keys for each design which he wants to share.

The first way is not proper method because all the designs may be leaked. For the second way efficiency is the major concern because it needs as many keys as the number of designs he want to send to the next person. So transferring these keys securely and storing them requires very expensive secure storage, which is very heavy and expensive. For the above problem the best solution is that Alice encrypts files with different public key but

sends a single constant size aggregate key to decrypting the file. Here, the burden on network is reduced as the problem of sending all the corresponding keys is replaced by sending just a single aggregate key. The expenses of having a tamper proof storage are usually high. The cost of secured storage for storing these secret keys is also reduced by storing the aggregate key due to its small size.

V. PROPOSED SOLUTION

Previous work in secured data storage has focused on how to perform privacy preserving policies. A few works have focused on secure key management approaches. In the some constrains such as aggregation of key, constant size cipher text, secure storage of them have remained the most important challenges. To ignore such constraints, we propose a new scheme Key-Aggregation Method (KAM). This is secured and energy efficient way of implementing asymmetric cryptosystem for manufacturing data sharing in the cloud storage. It produces fixed-size encrypted texts and the flexibility to choose number of secret keys to be aggregated.

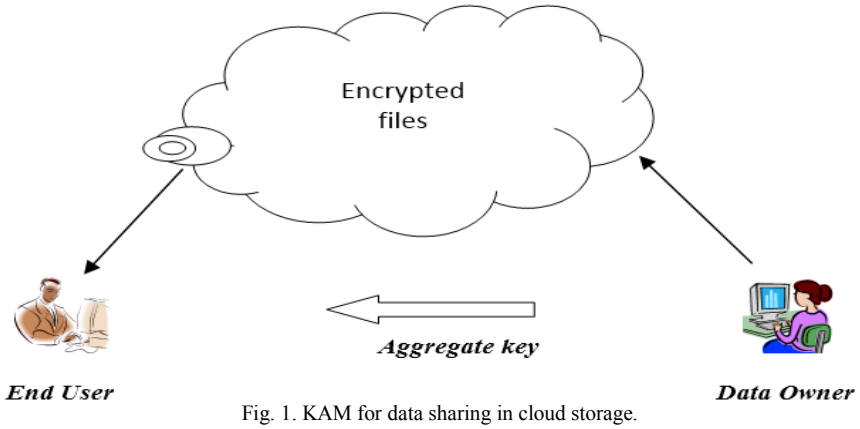
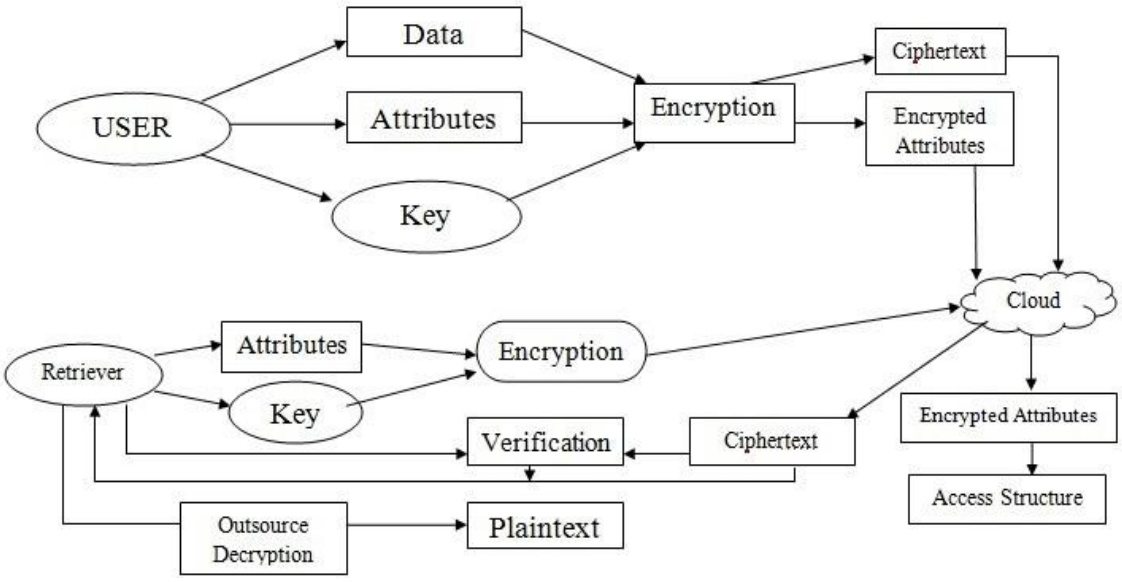


Fig. 1. KAM for data sharing in cloud storage.

In KAM, referring to Fig. 1. from [2], users encrypt the data using a public-key under an identifier of cipher text known as class. These cipher texts are actually categorized into separate classes. The owner of the key holds a master secret called master-secret key, which is used to obtain secret keys for different classes. The authorized user can decrypt only those cipher texts which he has the right to access.



This scheme has five basic algorithms and it is as follows:

Initial Setup phase (1l, n):

This is executed by the data owner to create an account on any untrusted server. The security level parameter and the number of ciphertext classes n is taken as input. The public system parameter param is given as output.

Generation of Key pair:

The data owner executes this algorithm for randomly generating a public/master-secret key pair (pk,msk).

Encryption phase (pk, i , m):

It is executed by the one who wants to encrypt the data. Public-key pk, an index i, corresponding to ciphertext class and a message m is taken as input. The ciphertext C is given as output.

Extraction (msk, S):

This is executed by the data owner for giving the decrypting power for certain set of ciphertext classes to the user. The master-secret key msk, and a set S of indices belonging to different classes is given as input.

The aggregate key for the set S is given as output i.e. KS.

Decryption (KS, S, i, C):

It is executed by the delegate who got the an aggregate key KS, the set S, an index I associating the cipher text class to which cipher text C belongs to. The output obtained will be the message m if I belong to S. Most importantly, the extracted key can be an aggregate key which will be as compressed as a secret key for a single class, but encompasses the decryption power for any subset of cipher text classes. The cipher text key size and the decryption key size both are constant. A novel technique of aggregating the secret keys is proposed. In this schema one can aggregate as many number of secret keys and make them as compact as a single key, which has the power of all the keys aggregated in it. As data sharing is one of the prime functionality in cloud storage, the secured, efficient and flexible sharing of data is proposed. When compared to its compressing factor, it has the ability to compress the secret keys into an aggregate key which has same size as that of a single key. As it is public-key cryptosystem, it is the efficient technique which can be utilized.

VI. Conclusion and Future Work

Although there are many benefits to adopting cloud computing, there are also some significant barriers to adoption. Because cloud computing represents a new computing model, there is a great deal of uncertainty about how security at all levels (e.g., network, host, application, and data levels) can be achieved. That uncertainty has consistently led information executives to state that security is their number one concern with cloud computing. Using this proposed scheme of key management, multiple files can be extracted by using single constant-size key at the same time. This paper presents a secure method of data storage in manufacturing cloud. The goal of the framework is to ensure data integrity and data confidentiality. In this method the number of key size will be reduced, it also reduces storage space to store this aggregate key. The files which are out of the requested set of data will remain confidential. The ability of cloud computing to adequately address privacy regulations has been called into question. [14] Organizations today face numerous different requirements attempting to protect the privacy of individuals' information, and it is not clear (i.e., not yet established) whether the cloud computing model provides adequate protection of such information, or whether organizations will be found in violation of regulations because of this new model.

REFERENCES

[1] L.M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner. A break in the clouds: towards a cloud definition, in: ACM SIGCOMM
[2] Computer Communication Review, 2008.p.50-55.
[3] M.B. Mollah, K.R. Islam, and S.S. Islam. Next generation of computing through cloud computing technology, in: 2012 25th IEEE Canadian.
[4] Conference on Electrical Computer Engineering (CCECE), May 2012.p.1-6.
[5] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou. Achieving secure, scalable and fine-grained data access control in cloud computing, in: IN-FOCOM, 2010 Proceedings IEEE, 2010.p.1-94.
[6] Jens-Matthias Bohli, Nils Gruschka, Meiko Jensen, Luigi Lo Lacono and Ninja Marnau, "Security and Privacy-Enhancing Multicloud Architectures", IEEE Transactions on Dependable and Secure Computing, VOL. 10, NO. 4, JULY?AUG 2013
[7] Arun Kumar S, S. Dhanasekar , "A Literature Survey on Key Aggregation System for Secure Sharing of Cloud Data", International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 3, Issue 12, December 2014.
[8] Gangolu Sreedevi, Prof. C. Rajendra," ICCC: Information Correctness to the Customers in Cloud Data Storage", in the year of June 2012.
[9] Rupali Sachin Vairagade, Nitin Ashokrao Vairagade," Cloud Computing Data Storage and Security Enhancement", in the year of August 2012.
[10] Kalpana Batra, Ch. Sunitha, Sushil Kumar," An Effective Data Storage Security Scheme for Cloud Computing", in the year of June 2013.
[11] K. Rajasekar & C. Kamalanathan," Towards of Secured Cost-Effective Multi-Cloud Storage in Cloud Computing", in the year of 2012.
[12] Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing".
[13] Mehdi Hojabri," Ensuring data storage security in cloud computing with effect of Kerberos", in the year of July 2012.
[14] S. Roy, M. Conti, S. Setia, and S. Jajodia, "Secure data aggregation in wireless sensor networks," IEEE Trans. Inf. Forensics Security, vol. 7, no. 3, pp. 1040–1052, Jun. 2012.
[15] S. Roy, S. Setia, and S. Jajodia, "Attack-resilient hierarchical data aggregation in sensor networks," in Proc. ACM Workshop Security Sensor Adhoc Netw. (SASN), 2006, pp. 71–82.
[16] 694 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 9, NO. 4, APRIL 2014
[17] M. B. Greenwald and S. Khanna, "Power-conservative computation of order-statistics over sensor networks," in Proc. 23th SIGMOD Principles Database Syst. (PODS), 2004, pp. 1–11.
[18] P. Flajolet and G. N. Martin, "Probabilistic counting algorithms for data base applications," J. Comput. Syst. Sci., vol. 31, no. 2, pp. 182–209, 1985.
[19] L. Hu and D. Evans, "Secure aggregation for wireless networks," in Proc. Workshop Security Assurance Ad Hoc Netw., 2003, pp. 384–391.
[20] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in Proc. ACM Conf. Comput. Commun. Security (CCS), 2006, pp. 278–287.
[21] B. Chen and H. Yu, "Secure aggregation with malicious node revocation in sensor networks," in Proc. 31st Int. Conf. Distrib. Comput. Syst. (ICDCS), 2011, pp. 581–592.
[22] D. Wagner, "Resilient aggregation in sensor networks," in Proc. ACM Workshop Security Sensor Adhoc Netw. (SASN), 2004, pp. 68–79.
[23] L. Buttyan, P. Schaffer, and I. Vajda, "Resilient aggregation with attack detection in sensor networks," in Proc. 2nd IEEE Workshop Sensor Netw. Syst. Pervasive Comput., Mar. 2006, pp. 331–336.
[24] B. Przydatek, D. Song, and A. Perrig, "SIA: Secure information aggregation in sensor networks," in Proc. 1st Int. Conf. Embedded Netw. Sensor Syst. (SenSys), 2003, pp. 255–265.
[25] K. Frikken and J. A. Dougherty, "An efficient integrity-preserving scheme for hierarchical sensor aggregation," in Proc. 1st ACM Conf. Wireless Netw. Security (WiSec), 2008, pp. 68–76.
[26] S. Nath, H. Yu, and H. Chan, "Secure outsourced aggregation via one- way chains," in Proc. 35th SIGMOD Int. Conf. Manag. Data, 2009, pp. 31–44.
[27] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor networks," in Proc. Int. Conf. Mobile Comput. Netw. (Mobi COM), 2001, pp. 189–199.
[28] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. F. Abdelzaher, "PDA: Privacy-preserving data aggregation in wireless sensor networks," in Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM), May 2007, pp. 2045–2053.
[29] S. Roy, M. Conti, S. Setia, and S. Jajodia. (2013). Secure Data Aggregation in Wireless Sensor Networks: Filtering out the Attacker's Impact.