International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 1 | ISSN : 2456-3307

Key Management Scheme based on the Attribute-Based Encryption for Sensor Networks

Bhattu Harikrishna[1], Dr. V. B. Narashima[2]

[1]Research Scholar, Department of Computer Science, University college of Engineering, Osmania University, Hydrabad, Telagana, India

[2]Assitant Professor, Department of Computer Science, University college of Engineering, Osmania University, Hydrabad, Telagana, India

ABSTRACT

Time and Trend has its own way of approach to the technological research. Recently, Information Technology and its implication play the most important role in the village area for the context of the taking consideration to industry of information data provenance needs to be secured since it may reveal private information about the sensitive data while the cloud service provider does not guarantee confidentiality of the data stored in dispersed geographical locations. Therefore, in addition to provide protection to the sensitive data, it is vital to make the data provenance secure. Another motivation to secure data provenance is for providing the enforceability and non-repudiation. The method investigates the problem of securing data provenance in the cloud and proposes a scheme that supports encrypted search while protecting confidentiality of data provenance stored in the cloud. In the proposed method, we consider a cloud data system consisting of data owners, data users, cloud server and third-party auditor. Initially the proposed method has three phases namely setup phase, key generation phase and storage phase. At first setup phase the data owner is reveal their information. Next key generation phase, in this phase the data owner get their key. If the data owner stores the sensitive data to cloud server, the data owner is encrypt their document in our proposed method we use double encryption technique to encrypt the user document. Here RSA and Blowfish algorithm is used to encrypt the document with high security. After the encryption the document stored in the cloud along with an access structure that specifies which types of user are allowed to access the document.

Keywords: Attribute-Based Encryption, Access Control, Outsourcing Computation, Key Issuing, Communication Cost Distribution, Optimized Matrix

I. INTRODUCTION

In the history of cryptology up to 1975, encryption and decryption (the reverse of encryption) algorithm of all the cryptosystems employ the same key. This means the encryption key, held by the principal who encrypts a message, and the decryption key, held by the one who will receive the and decrypt it, are the same. Taking Caesar cipher as an example, we can consider the number of positions shifted in the plaintext as both the encryption key and the decryption key. Such kind of cryptosystems is known as symmetric-key cryptosystem. The symmetric nature of the encryption and the decryption keys requires a key to be agreed upon by the two communicating parties by some possibly non-cryptographic means; for example, a face-to-face meeting, such that no one else knows any part of the key. In other words, a prior shared secret should be established by an authenticated and private communications channel before a cryptosystem can be used. In practice, various difficulties may arise to distribute keys.

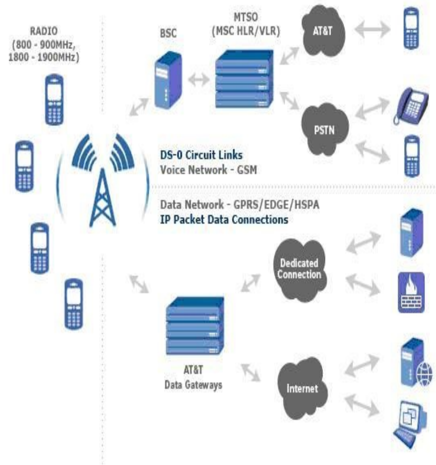

Figure 1.1. Illustration of the Sensor Network

After all, the task of establishing a secret is closely related to the original goal of an encryption system. In 1975, protocol for establishing a shared secret-key over a public (but still authenticated) communication channel, without using any prior shared secret. Eavesdroppers can still read the transcripts of communication generated during the execution of the protocol, but cannot derivate the session key that the protocol participants compute locally and secretly. The protocol is now known key exchange, key exchange as suggested by Hellman in 2002, in recognition. Describe how existing work provides limited solutions to address some of them.

II. RELATED WORK

This discovery was considered as a brand new concept in the field which was named as public-key cryptography, where different can be made public. Public-key cryptosystems are also referred to as asymmetric cryptosystems. Key establishment: All nodes in the network use the same traffic encryption key (TEK) to communicate securely with each other. The TEK is derived as follows in two phases. In the first phase, the network starts with no clusters at all. Then every node broadcasts its weight and when a node finds its weight to be greater than its neighbors', it announces itself as the cluster head, while its neighbors become cluster members. The formation of the cluster is complete after the cluster head distributes a cluster key to its cluster members. The first phase ends with the cluster heads forming a backbone of the network. In the second phase, the cluster heads that have a larger weight than its neighboring cluster heads elect themselves as the potential key managers (PKM). After an exponential back off period, a PKM generates (using an unspecified algorithm) and distributes a TEK to other cluster heads. More than one PKM might generate a TEK at the same time, so a non-PKM cluster head might receive more than one TEK. Simulate the behavior of the Bell- model using a single role hierarchy. In particular, it is not possible to support the assignment of "mixed" permissions in existing work, which include both read and write access to resources. Furthermore, no work has studied the simulation of the more complex version of the model that includes the current security function, or provided support for limited discretionary features of the model in role-based models.

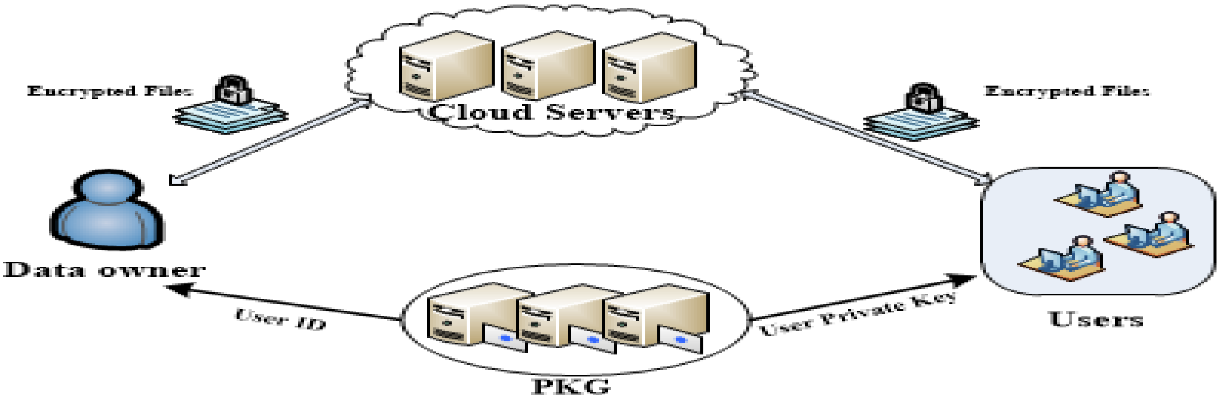

Figure 2.1 Illustration of Group Based Encrypted Key Cloud Cluster Node

To increase control, we argue that extensibility and customizability should be built into public clouds. Specifically, clouds such as Amazon S3, Google Docs, Face book, and the DHT should allow their clients to customize data management properties, such as data placement, availability of forensic logs, and replication schemes. This paper takes a first step toward the realization of the extensible-cloud vision, proposing the design, implementation, and evaluation of Comet, an extensible distributed storage system. While motivated broadly by the inflexibility in today's Web clouds, Comet focuses on a particular kind of service – a distributed key/value store based on peer-to-peer DHTs. Comet's design is informed by our experience building Vanish on top of the inflexible DHT. We begin by describing this context and motivate the need for extensible DHTs.

III. METHODOLOGY

A user wants to access the document, at first verify their signature if the verification success the user continue to access a set of documents. If some dispute arises in a stored document, the TPA can track the dishonest user's identity as follows. The TPA first gets the signature from the cloud server the identity can be pinpointed. The proposed provenance system can efficiently add or revoke users. The data owner request with the user identity to the cloud server and asks the cloud server to include the new user for encryption. To revoke the user, the data owner send request to update the access structure to cloud server. To reduce the computational cost of the proposed method, here we have to split the information and stored in the two different servers. After that, we have applied greedy selection, to find the optimal results from the two servers. The confidentiality of documents can be derived directly because of the secure double encryption scheme.

Key deployment: Every node is imprinted with *k* keys chosen at random from the key pool of size *P*.

- **Key establishment:** Only node-to-node communications are supported. If the two nodes share at least a key, session keys are derived from the shared key. If the two nodes do not share any key, but have secure links to a common neighbor, then the two nodes can establish a session key through their common neighbor acting as a trusted third party. Note the two nodes in this context have to be within radio range, otherwise neighbor discovery cannot take place.
- **Node addition:** When a node is added to the network, the node broadcasts a list of identifiers that identify the keys it has. The neighbors reply with their lists of key identifiers. By comparing the lists, the new node and its neighbors discover what keys they share. Session keys are then derived from the shared keys.
- **Node eviction:** To evict a node, a controller first broadcasts a message containing a signed list of *k* key identifiers for the key ring to be revoked. Then the controller unicasts to each node *A* the signing key encrypted with key *KA*, where *KA* is derived from the keys the controller shares with.

The subsequent IBE cipher text is marked utilizing the signing key relating to the verification key. Cipher text is splitted into three parts, the confirmation key, the IBE cipher text and the signature. For decoding a cipher text, receiver first understand that the signature on the IBE cipher text regarding the check key is substantial. Unnamed secret key issuing in ID based cryptosystems was generally considered by Sui et al., in a framework where the duties of verification and key issuing are isolated to local registration experts and the KGC. Rather than having a LRA to issue a signature, a user supplies a secret key to the LRA. To utilize their convention, a LRA is required to send a list of identities and passwords to the KGC, while this protocol does not require any correspondence amongst them and this prerequisite isn't appropriate for our motivation. Moreover their anominity be planned an ad-hoc security analysis conducted after that it would be sent and considered as the secure so long as nobody found an attack against it. Basically in more complex system the more secure scheme were assumed. If attack was found and user is aware of attack then the system was redesigned and redeployed in a novel scheme.

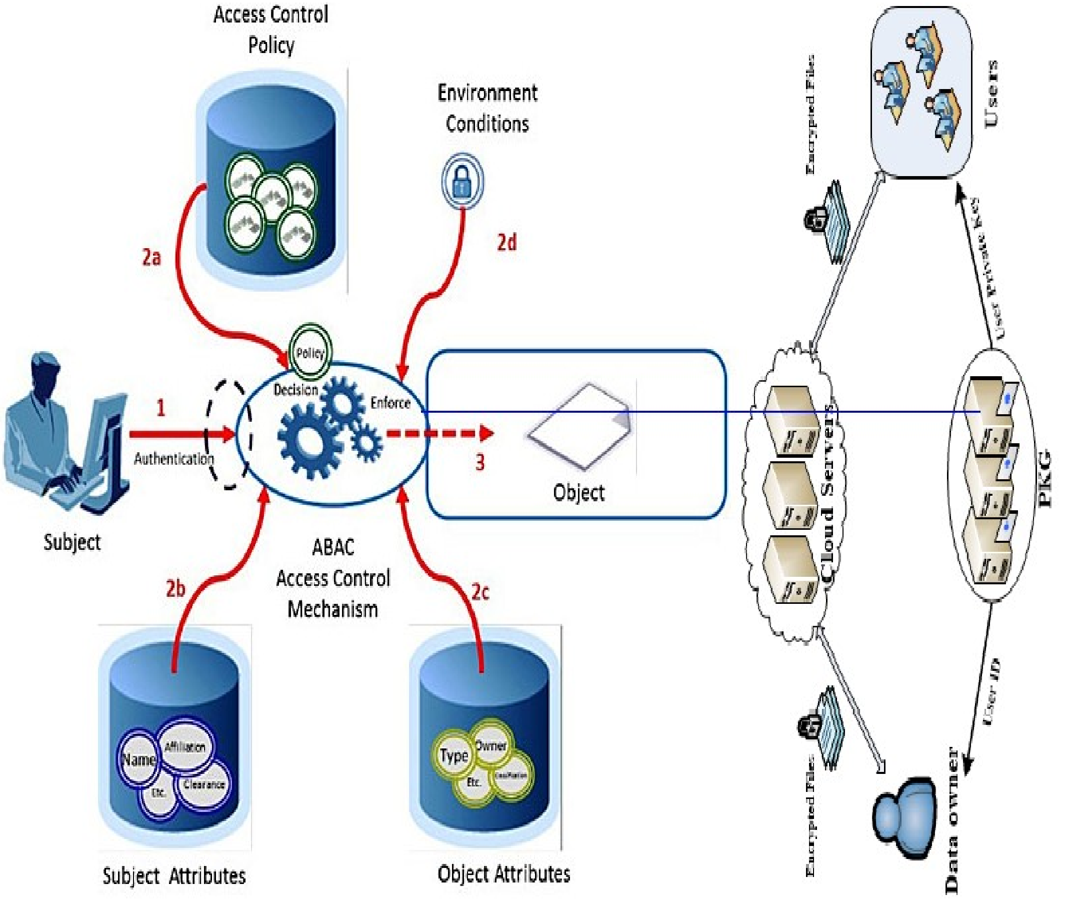

Figure 3.1. Schema Key Based Node Architecture Model

The paging subsystem presents a translation layer between the virtual addresses used by software running on the processor and the physical addresses corresponding to the actual location in memory that is being referenced. Each block of virtual memory can be mapped to an arbitrary block of physical memory, with the exact mapping being maintained as an entry in the page table. The processor restricts the ability to update the page table to only privileged code. The OS kernel is responsible for maintaining this mapping (for the moment, we will assume a hypervisor is not present). When an application is being run, the page table mappings are configured to allow access only to physical memory allocated to the currently running application. Because the mapping is controlled by the operating system, the application is restricted from accessing memory belonging to other applications on the system. While the OS kernel shares the same page table as the application, memory associated with the kernel remains protected through a privilege level bit enforced by the processor.

3.1 Evaluation and Analysis

Unidentified has pulled in consideration for the security benefits, and as a use to develop public key encryption with keyword search as follows. For representing the keywords identity strings are used. For testing whether the cipher text is labeled with a specific keyword or not the private key is used. Role performed by the KGC is also known as trapdoor generation. The procedure for generating the encoded tag is that encrypt an arbitrary message by utilizing the keyword considered as identity in IBE and append the message with that tag. To find the cipher text labeled with a keyword, one tries to utilize a trapdoor to decode the tag, and check whether the outcome coordinates the accompanying with message. In proposed scheme, ACI KGC understands that the cooperation of the private key does not issue the keyword from an encoded tag. In an anonymous credential system, clients wish to acquire and demonstrate ownership of credentials while remaining anonymous. In such work it is expected that each client has an unique secret key and there are distinct recommendations for how to demonstrate that a given key is valid and to prevent users from loaning out their keys. At that point the client can interface with each authority under an alternate pseudonym such a path, to the point that it is difficult to link different pseudonyms to a similar client. In the meantime, all of a user's pseudonyms, the subsequent credentials, are attached to a similar secret key with the goal that the client can demonstrate that he has both trait set A from one expert and set B from another. These systems show that, with carefully crafted abstractions and mechanisms, users can regain control over various aspects of their data without losing the new technologies' advantages. is privileged. A specific page of memory can be indicated as read-only for any user code running but read/write for privileged code. The OS kernel uses this privileged bit and associated access restrictions in the page tables to prevent user code from writing to ages belonging to the operating system.

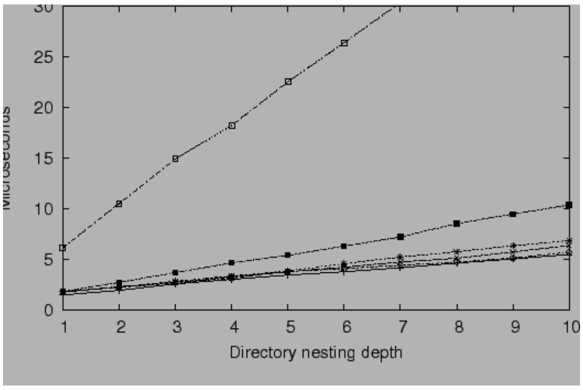

Figure 3.2.1 Evaluation Graph of Attribute and Time

Any attempt by user code to either execute privileged assembly instructions or modify read-only pages of memory is trapped by the processor and forwarded to the privileged code. The privileged code can either reject the attempt or emulate the operation on behalf of the user code.

IV. CONCLUSION AND FUTURE WORK

The overarching goal of these techniques was to increase users' control over various aspects of their data in the cloud and on mobile devices. Keypad provides remote access auditing and control over data stored on stolen mobile devices; Vanish offers data lifetime control on the Web; Comet allows users to customize various data management properties in a storage cloud; and Menagerie allows users to regain a unified organizational view over their scattered Web data. In cryptography strength is the property of a public key encryption technique such that decoding a cipher text generated with a public key is failed when decoding with a private key matching to another public key. Practically this technique is accomplished by using the appropriate padding technique. Moreover there is nothing essential in the regular security descriptions to incorporate such functionality.

V. REFERENCES

[1]. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.

[2]. S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc.Int'l Conf. Financial Cryptography and Data Security (FC), pp. 136-149, Jan. 2010.

[3]. S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing," Proc. IEEE INFOCOM, pp. 534-542, 2010.

[4]. M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc. USENIX

Conf. File and Storage Technologies, pp. 29-42, 2003.

[5]. E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius: Securing Remote Untrusted Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 131-145, 2003.

[6]. G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 29-43, 2005.

[7]. R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010.

[8]. B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," Proc. Int’l Conf. Practice and Theory in Public Key Cryptography Conf. Public Key Cryptography, http://eprint.iacr.org/2008/290.pdf, 2008.

[9]. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security (CCS), pp. 89-98, 2006.

[10]. D. Naor, M. Naor, and J.B. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers," Proc. Ann. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-62, 2001.

[11]. D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 213-229, 2001.

[12]. D. Boneh, X. Boyen, and H. Shacham, "Short Group Signature," Proc. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-55, 2004.