Revocable IBE Scheme for Efficient Identity Based Encryption on Cloud Infrastructure

[1]M.T. Ramakrishna Raju, [2]Jeevana sujitha
[1,2]Dept. of CSE, SRKR Engineeringing College, Bhimavaram, AP, India

Abstract
Efficient revocation remains well examined in traditional PKI setting, nevertheless the cumbersome control of certificates is just the duty that IBE strives to help relieve. Inside this paper, striving at tackling the critical issue of identity revocation, we introduce outsourcing computation into IBE for the first a serious amounts of propose a revocable IBE plan inside the server-aided setting. Identity-Based File encryption (IBE) which simplifies everyone key and certificate management at Public Key Infrastructure(PKI) is a crucial choice to public key file encryption. However, the very best efficiency drawbacks of IBE could be the overhead computation at Private Key Generator (PKG) during user revocation. Our plan offloads most of the key generation related techniques during key-giving and key-update techniques obtaining a vital Update Cloud Company, departing just a ongoing amount of simple methods for PKG and clients to complete your geographical area. Finally, we provide extensive experimental results in exhibit the efficiency within our recommended construction. This goal is accomplished obtaining a manuscript collusion-resistant technique: we utilize a hybrid private key for each user, through which an AND gate is involved for linking and bound the identity component together with time component. Additionally, we use Triple DES algorithm for providing security to the documents that are stored in the cloud because the cloud service provider(KU-CSP) is not a trusted one.

Keywords
Identity-based Encryption, Revocation, Outsourcing, Cloud Computing

I. Introduction

Cloud computing, the long held dream of computing as a utility, has a potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way the IT industry is designed and purchased. Cloud computing provides facilities by which users can access the applications as utilities, over the internet. It can also refer to the delivery of computing resources over the internet. Cloud computing allows users to manipulate, configure and access online applications. Cloud can provide services over internet that is on public networks like WAN, LAN or VPN. There are three service modules on which the cloud computing is based. They are Infrastructure as a Service (Iaas), Platform as a Service (PaaS) and Software as a Service (SaaS). IaaS provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc. PaaS provides the runtime environment for applications, development and deployment tools, etc. SaaS model allows using software applications as a service to the end users. Content delivery networks are based on the Network as a Service (NaaS).

II. Related work

Receiver will obtain the non-public key connected when using the corresponding identity from Private Key Generator (PKG) has the ability to decrypt such cipher text. Therefore, sender using IBE don't have to research public key and certificate, but directly encrypts message with receiver's identity. Identity-Based encryption (IBE) is clearly a fascinating option to public key encryption, that's recommended to simplify key management inside the certificate-based Public Key Infrastructure (PKI) through the use of human-intelligible particulars as public keys. Though IBE enables an arbitrary string since the public key that's considered just like a beautiful edge over PKI, it requires a dependable revocation mechanism. Particularly, once the private keys of some clients get compromised, we must provide a mean to revoke such clients from system. In PKI setting, revocation mechanism is identified by appending validity periods to certificates or using involved mixtures of techniques. Nevertheless, the cumbersome control of certificates is just the responsibility that IBE strives to help relieve. However, this mechanism would create a overhead load at PKG. In another word, all the clients whether their keys are actually revoked otherwise, need to reference to the PKG periodically to show their particulars increase new private keys. It requires that PKG is web the secure funnel must be maintained for individual's transactions, this is a bottleneck for IBE system as the quantity of clients evolves. Therefore, key-update efficiency at PKG has the ability to be significantly reduced from straight line for your height of people binary tree. Nevertheless, we explain that even though the binary tree introduction is able to do get yourself a relative high finish. Along with introduction of cloud-computing, there's emerged the ability for clients to buy on-demand computing from cloud-based services for instance Amazon's EC2 and Microsoft's Home windows Azure [1]. So that it desires a completely new working paradigm for showing such cloud services into IBE revocation to repair of efficiency and storage overhead described above.

A naive approach must be to simply give you the PKG's master reaction to the Cloud Providers (CSPs). The CSPs could then simply just update all the private keys when using the traditional key update technique and transmit the non-public techniques of unrevoked clients. However, the naive approach depends upon improper assumption the CSPs are fully reliable that's allowed to buy the specific key for IBE system [2]. However, used everybody clouds are likely outdoors inside the reliable domain of clients and they're curious for users' individual privacy. Due to this, challenging so that you can design an excellent revocable IBE intend to lessen the overhead computation at PKG by enabling a united nations reliable CSP is elevated. In this paper, we introduce outsourcing computation into IBE revocation, and formalize the security concept of outsourced revocable IBE the first time to great our understanding.

The advise a concept to offload all the key generation related techniques during key-giving and key-update, departing only constant volume of simple method of PKG and qualified clients to accomplish your geographical area. Inside our plan, such as the suggestion in, everyone knows revocation through upgrading the non-public keys within the unrevoked clients [4]. But unlike realistically work which trivially concatenates time period with identity for key generation/update and needs to re-issue the whole

private key for unrevoked clients.
To advice a manuscript collusion-resistant key giving technique: we use a hybrid private key for each user, through which an AND gate is involved helping you to connect up and bound two sub-components, namely their entity component together with time component [3]. Initially, user has the capacity to provide the identity component plus a default time component and includes the most effective output as extended due to the available one server that follows the suggested protocol.

III. Problem Statement

The available scheme offloads most of the key generation related operations during key-issuing and key-update processes to a Key Update Cloud Service Provider, leaving only a constant number of simple operations for PKG and users to perform locally. This goal is achieved by utilizing a novel collusion-resistant technique: we employ a hybrid private key for each user, in which an AND gate is involved to connect and bound the identity component and the time component. Moreover, we remark that it cannot trivially utilize an identical updated time component for all users because revoked user is able to re-construct his/her ability through colluding with unrevoked users. To eliminate such collusion, we randomly generate an outsourcing key for each identity ID, which essentially decides a "matching relationship" for the two sub-components. Furthermore, we let KU-CSP maintain a list to record user's identity and its corresponding outsourcing key. In key-update, we can use OKID to update the time component TK[ID]T for identity ID But the scheme fails to provide security in all aspects and takes a lot of time.

IV. Proposed Methodology

Previously our revocable scheme uses DES for providing security to the data. Even though it is faster it fails to provide security in case of un trusted KU-CSP. So in our proposed methodology we are using Triple DES algorithm.

Triple DES is another mode of DES operation. It takes three 64-bit keys, for an overall key length of 192 bits. In Stealth, you simply type in the entire 192-bit (24 character) key rather than entering each of the three keys individually. The Triple DES DLL then breaks the user-provided key into three subkeys, padding the keys if necessary so they are each 64 bits long. The procedure for encryption is exactly the same as regular DES, but it is repeated three times, hence the name Triple DES. The data is encrypted with the first key, decrypted with the second key, and finally encrypted again with the third key.

Triple DES runs three times slower than DES, but is much more secure if used properly. The procedure for decrypting something is the same as the procedure for encryption, except it is executed in reverse. Like DES, data is encrypted and decrypted in 64-bit chunks. Although the input key for DES is 64 bits long, the actual key used by DES is only 56 bits in length. The least significant (right-most) bit in each byte is a parity bit, and should be set so that there are always an odd number of 1s in every byte. These parity bits are ignored, so only the seven most significant bits of each byte are used, resulting in a key length of 56 bits. This means that the effective key strength for Triple DES is actually 168 bits because each of the three keys contains 8 parity bits that are not used during the encryption process.

VI. Result Analysis

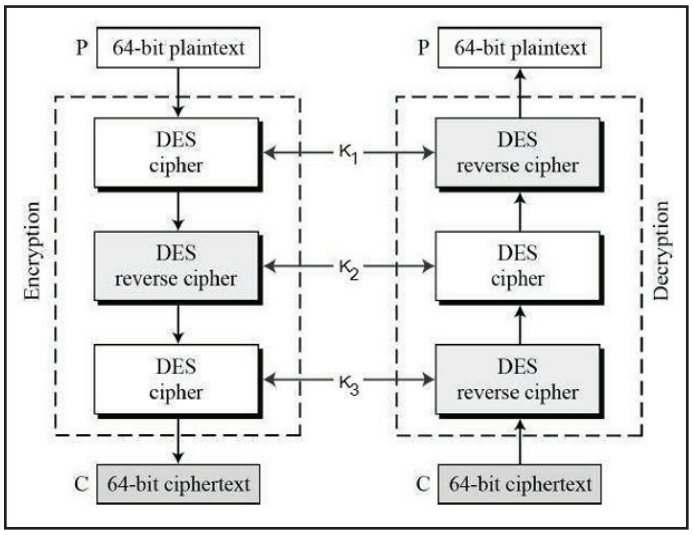

Fig. 1: Three Key Triple DES Model

Before using 3TDES, user will first generate and distribute a 3TDES key K, which consists of three different DES keys K_1, K_2 and K_3. This means that the actual 3TDES key has length 3×56 = 168 bits. The encryption scheme is illustrated as follows −

The encryption-decryption process is as follows:

- Encrypt the plaintext blocks using single DES with key K_1.
- Now decrypt the output of step1 using single DES with key k_2.
- Finally encrypt the output of the step2 using single DES with key k_3.
- The output of the step3 is cipher text.
- Decryption of the cipher text is a reverse process. User first decrypts k3, then encrypts with k2 and finally decrypt with k1.

Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for single DES by setting K1, K2, and K3 to be the same value. This provides backwards compatibility with DES.
Second variant of Triple DES (2TDES) is identical to 3TDES except that K3is replaced by K1. In other words, user encrypts plaintext blocks with key K1, then decrypt with key K2 and finally encrypt with K1 again. Therefore, 2TDES has a key length of 112 bits backwards compatibility with DES.
Triple DES systems are significantly more secure than single DES, but these are clearly a much slower process than encryption using single DES.

V. Conclusion and Future Work

The scheme offloads most of the key generation related operations during key-issuing and key-update processes to a Key Update Cloud Service Provider, leaving only a constant number of simple operations for PKG and users to perform locally. This goal is achieved by utilizing a novel collusion-resistant technique: we employ a hybrid private key for each user, in which an AND gate is involved to connect and bound the identity component and the time component. Moreover, we remark that it cannot trivially utilize an identical updated time component for all users because revoked user is able to re-construct his/her ability through colluding with unrevoked users. To eliminate such collusion, we randomly generate an outsourcing key for each identity ID, which essentially decides a "matching relationship" for the two

sub-components. Additionally, we use Triple DES algorithm for providing security to the documents that are stored in the cloud because the used KU-CSP is not a trusted one.

References

[1] M. Green, S. Hohenberger, B. Waters,"Outsourcing the decryption of ABE ciphertexts," [Online] Available: https://www.usenix.org/legacy/event/sec11/tech/full_papers/Green.pdf.

[2] Z. Zhou, D. Huang,"Efficient and secure data storage operations for mobile cloud computing," [Online] Available: http://mdatechsys.com/project/Mobile%20Computing/Efficient%20and%20Secure%20Data%20Storage%20Operations%20for.pdf.

[3] D. Boneh, X. Ding, G. Tsudik, C. Wong,"A method for fast revocation of public key certificates and security capabilities," [Online] Available: http://crypto.stanford.edu/~dabo/pubs/papers/sem.pdf

[4] B. Libert, J.-J. Quisquater,"Efficient revocation and threshold pairing based cryptosystems," [Online] Available: https://www.researchgate.net/publication/221343635_Efficient_revocation_and_threshold_pairing_based_cryptosystems

[5] M. J. Atallah, K. Pantazopoulos, J. R. Rice, E. E. Spafford, "Secureoutsourcing of scientific computations," [Online] Available: https://www.cs.purdue.edu/homes/jrr/pubs/AdvComp.pdf

[6] C. Wang, K. Ren, J. Wang,"Secure and practical outsourcing of linear programming in cloudcomputing," [Online] Available: Online: https://pdfs.semanticscholar.org/6b8e/dd0a6631ea436d7f178e69050e983d189431.pdf

[7] B. Libert, D. Vergnaud,"Adaptive-id secure revocable identity based encryption," In Topics in Cryptology (CT-RSA˝09), [Online] Available: http://www.di.ens.fr/~vergnaud/publis/ct-rsa09.pdf

[8] S. Yu, C. Wang, K. Ren, W. Lou,"Attribute based data sharing with attribute revocation," [Online] Available: http://www.cs.cityu.edu.hk/~congwang/papers/ASIACCS10-sharing.pdf

[9] B. Zhang, J. Wang, K. Ren, C. Wang,"Privacy-assured outsourcing of image reconstruction service in cloud," [Online] Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6562794