ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.6

Efficient Identity-Based Encryption used in Cloud Computing

Priyanka .M. Narwade[1], Prof. S. P. Kosbatwar[2]
PG Student[1], Associate Professor[2]
Department of Computer Engineering
STES's Smt. Kashibai Navale College of Engineering, Pune, India

Abstract:
In public key encryption every user must have a pair of keys, public key and private key, for encrypting and decrypting messages. An Identity-based encryption (IBE) eliminates the need for a Public Key Infrastructure (PKI). IBE uses the human intelligible identities (e.g., unique name, email address, IP address, etc) as public keys. The private keys of users are obtained from a trusted third party called as Private Key Generator (PKG). This paper proposes the cloud based revocable identity-based proxy re-encryption scheme that supports user revocation but also delegation of decryption rights. At the end of the given time period cloud acting as a proxy will re-encrypt all ciphertext of the user under the current time period to the next time period. Revoked users cannot decrypt the ciphertexts by using the expired private key anymore.

Keywords: Cloud computing, identity-based encryption, outsourcing, public key encryption, re-encryption, revocation.

I. INTRODUCTION

Identity-Based Encryption (IBE) is proposed to simplify the key management in the certificate based Public Key Infrastructure (PKI). IBE is an interesting alternative to public key encryption where it uses human-intelligible identities (e.g., unique name, email address, IP address, etc) as public keys. Therefore, the sender using IBE does not need to look for the public key and corresponding certificate of the receiver, rather it directly encrypts message with the receivers identity. Accordingly, receiver obtaining the private key associated with corresponding identity from Private Key Generator (PKG) is able to decrypt such ciphertext. Though IBE provides advantages over public key encryptionit must provide some efficient revocation mechanism. Specifically, if the private keys of some users get compromised, we must provide a mean to revoke such users from system. In PKI, revocation mechanism is realized by appending time component to the certificate or using involved combination techniques [1][2][3]. If a user is revoked, the certificate authority will add his/her certificate to a certificate revocation list (CRL). Anyone who wants to send a encrypted message for this user hatcheck the certificate of the user against the CRL. If the certificate is on the list, the sender knows that this user has been revoked and therefore, will not further share any sensitive information with him/her. Unlike PKI, in identity-based encryption (IBE) scheme there is no such certificate. The motivation for IBE is to solve the problems related to certificate management. The first time in 2001 Boneh and Franklin [4] suggested the Identity-based encryption scheme, in which users update their private key periodically and sender encrypts message using receivers' identity concatenated with current time period. Every user in the system has to get in contact with PKG periodically to get new private key. For such transactions PKG must be online all the time, and a secure channel must be established between the PKG and users, for a very large number of users this will become a bottleneck. In 2008, Boldyreva, Goyal, Kumar [5] presented an IBE with efficient revocation scheme. In order to avoid the need for interaction and a secure channel between PKG and users they suggested the PKG may encrypt new keys of non-revoked users under their identities and the previous time period, and send the ciphertext to these users (or post them online). With this approach, for every non-revoked user in the system, the PKG is required to perform one key generation and one encryption operation per key update.

II. PRELIMINARY

This section review the definition of bilinear maps, one way identity-based encryption and identity-based encryption.

A. BILINEAR MAPS

My review on bilinear maps, using the following standard notation [4] [9] [11]:

1. G and G_T are two (multiplicative) cyclic groups for prime order p;
2. g is a generator of G.
3. $e : G \times G \rightarrow G_T$ is a bilinear map.

Let G and G_T be two groups as above. A bilinear map is a map $e : G \times G \rightarrow G_T$ with the following properties:

1. Bilinear: for all u, v $\in G$ and a, b $\in$ Z, we have $e(u^a, v^b) = e(u, v)^{ab}$.
2. Non-degenerate: $e(g, g) \neq 1$.

The G is a bilinear group if the group action in G can be computed efficiently and there exists a group G_T and an efficiently computed bilinear map $e: G \times G \rightarrow G_T$ as above. Note that $e(,)$ is symmetric since $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

B. ONE WAY IDENTITY-BASED ENCRYPTION

One can define an even weaker notion of security called as One Way Encryption (OWE) [12]. One way encryption is a weak notion of security since there is no way of preventing adversary from learning half the bit of plaintext. Hence, one way encryption schemes do not generally provide secure encryption. By adding private key extraction queries to the definition, one can extend the notion of one way encryption to the identity based systems.

C. IDENTITY-BASED ENCRYPTION

I give a brief review on IBE scheme which typically involves two entities named as PKG and users. IBE follows the four algorithms:

1) Setup (λ): The setup algorithm run by the key authority, which takes security parameter λ as input and outputs the public parameters PK and master key MK.

2) KeyGen (MK, ID): The private key generation algorithm run by the PKG, which takes as input the master key MK and user's identity ID $\in\{0,1\}^*$ and outputs the private key SK_{ID} corresponding to the identity ID.

3) Encrypt (M, ID'): The encryption algorithm run by the sender, which takes as input a message M and receivers' identity ID'. It outputs the ciphertext CT.

4) Decrypt (CT, $SK_{ID'}$): The decryption algorithm run by the receiver, which takes as input the ciphertext CT and her/his private key $SK_{ID'}$). It outputs the message M or an error $\perp$.

An IBE scheme must satisfy the consistency conditions. Specifically, when the private key SK_{ID} generated by algorithm KeyGen when an ID is given as input, then Decrypt (CT, SK_{ID}) = M where CT = Encrypt (M, ID).

III. SYSTEM MODEL

Figure. 1. Presents the system model for outsourced revocable IBE scheme as given in [7].

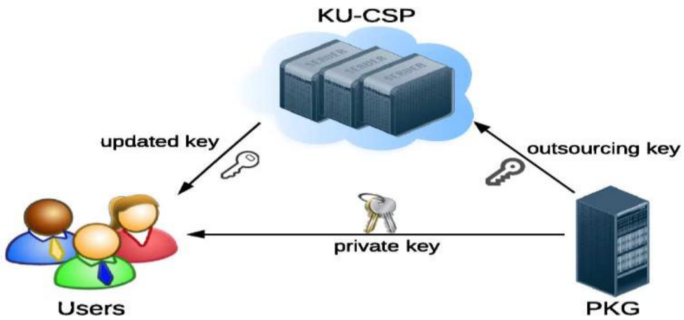

Figure.i. System model for ibe with outsourced revocation [7]

The KU-CSP can be considered as a public cloud run by a third party to serve the basic computing capabilities or services to the PKG over the network. The function of revocation is done by the KU-CSP, if a user is compromised then KU-CSP will revoke such user. As the KU-CSP is envisioned as a public cloud, it is hosted away from either PKG or users. KU-CSP provides a way to temporary extension to infrastructure, which reduce the PKG computation and storage cost. In this model initially users get in contact with PKG to obtain the private key, and after that for updating the key users get in contact with KU-CSP periodically. The key generation algorithm, run by PKG, outputs the private key for user and an outsourcing key for KU-CSP. For each unrevoked user KU-CSP updates the part of the private key i.e. a lightweight component of the private key.

IV. IMPLEMENTATION

This section presents the construction for outsourced revocable IBE scheme based on [6] [7] as follows.

A. Setup(λ):

The setup algorithm is run by PKG. It selects a random generator $g \in_R G$ as well as a random integer $x \in_R Z_q$, and sets $g_1 = g^x$. Then, PKG picks a random element $g_2 \in_R G$ and two hash functions H_1, H_2 : $\{0,1\}^* \rightarrow G_T$. Finally, output the public key PK= (g,g_1, g_2, H_1, H_2) and the master key MK = x.

B. KeyGen(MK, ID, RL, TL, PK):

The KeyGen algorithm is run by PKG. Whenever a new private key request arrives, PKG firstly checks whether the request identity ID exists in Revocation List (RL), if so the key generation algorithm is aborted. Next, PKG randomly selects $x_1 \in_R Z_q$ and sets $x_2 = x$ - x_1 mod q. It randomly chooses $r_{ID} \in_R Z_q$, and computes IK[ID] = ($g_2^{x_1} \cdot (H_1(\mathrm{ID}))^{r_{ID}}$, $g^{r_{ID}}$). Then PKG reads the current time period T_i from Time period List (TL). Accordingly, it randomly selects $r_{T_i} \in_R Z_q$ and computes TK $[ID]_{T_i}$ = ($d_{T_i 0}$, $d_{T_i 1}$), where $d_{T_i 0}$ = $g_2^{x_2} \cdot (H_2(T_i))^{r_{T_i}}$ and $d_{T_i 1} = g^{r_{T_i}}$. Finally, output SK_{ID} = (IK[ID], TK$[ID]_{T_i}$) and $OK_{ID} = x_2$.

C. DeKeyGen(SK_{ID}, T_i):

The DeKeyGen algorithm outputs a decryption key $SK_{ID|i}$ for the user ID under the timeperiod Ti or $\perp$ if ID has been revoked, where i $\in$ [1, poly(1^k)].

D. ReKeyGen($SK_{ID|i}$, MK, T_i, $T_{i'}$):

The ReKeyGen algorithm generates the re-encryption key as follows, where1≤ i <i'.

1. **ReKeyToken(MK, T_i , $T_{i'}$):**The ReKeyToken algorithm outputs a re-encryption key token$\varphi_{i\rightarrow i'}$.

2. **ReKey($SK_{ID|i}$, $\varphi_{i\rightarrow i'}$):** The ReKey algorithm outputs a re-encryption key $RK_{ID|i\rightarrow i'}$ which can be used to transform a ciphertext under (ID, T_i) to another ciphertext under (ID,$T_{i'}$).

E. Encrypt(*M, ID, T_i, PK):*

Suppose a user wants to encrypt a message M under identity ID and time period T_i. The sender selects a random value s $\in_R Z_q$ and computes C_0 =Me $(g_1, g_2)^s$, $C_1 = g^s$, E_{ID} = $(H_1(\mathrm{ID}))^s$ and E_{T_i} = $(H_2(T_i))^s$. Finally, publish the ciphertext as CT = (C_0, C_1,E_{ID}, E_{T_i}).

F. ReEnc($RK_{ID|i\rightarrow i'}$, C):

The ReEnc algorithm intakes $RK_{ID|i\rightarrow i'}$ and C under (ID, T_i) and outputs either a re-encrypted ciphertext C under (ID, $T_{i'}$) or a symbol $\perp$ indicating C is invalid, where 1≤ i <i'.

G. Decrypt(CT, SK_{ID},PK):

To decrypt a cipher text receiver computes:

$$M = \frac{C_0 e(d_1, E_{ID}) e(d_{T_i 1}, E_{T_i})}{e(C_1, d_0) e(C_1, d_{T_i 0})}$$

$$= \frac{M e(g_1, g_2)^s}{e(g, g_2)^{x_2 s} e(g, g_2)^{x_1 s}}$$

= M.

H. Revoke(RL, TL, $\{ID_{i_1}, ID_{i_2}, \ldots, ID_{i_k}\}$):

If users with identities in the set $\{ID_{i_1}, ID_{i_2}, \ldots, ID_{i_k}\}$ are to be revoked at time period T_i, PKG updates the revocation list as $RL' = RL \cup \{ID_{i_1}, ID_{i_2}, \ldots, ID_{i_k}\}$ as well as the time list through linking the newly created time period T_{i+1} onto original list TL. Finally send a copy for the updated revocation list RL' as well as the new time period T_{i+1} to KU-CSP.

I. KeyUpdate(RL, ID, T_{i+1}, OK_{ID}):

When KU-CSP receives a key-update request on ID, it firstly checks whether ID exists in the revocation list *RL* , if so KU-CSP returns $\perp$ and key-update aborted. Otherwise, KU-CSP fetches the corresponding entry (ID, $OK_{ID} = x_2$) in the user list *UL* . Then, it randomly selects $r_{T_{i+1}} \in_R Z_{q'}$ and computes $d_{T_{i+1}0} = g_2^{x_2} \cdot (H_2(T_{i+1}))^{rT_{i+1}}$ and $d_{T_{i+1}1} = g^{rT_{i+1}}$. Finally, output $TK[ID]_{T_{i+1}} = (d_{T_{i+1}0}, d_{T_{i+1}1})$.

V. PERFORMANCE EVALUATION

This section compare the proposed approach with IBE with revocation approach with the Identity-based encryption with Outsourced Revocation in Cloud Computing [3].

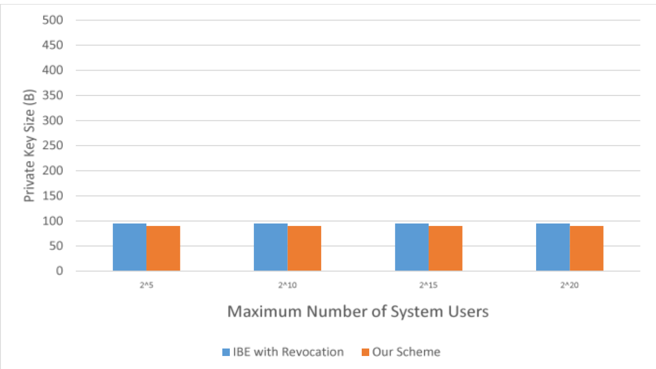

Figure.2. Comparison of private key size

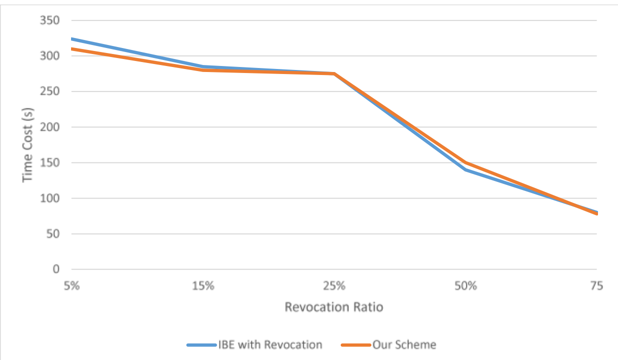

Figure.3. Comparison of key update at ku-csp

Below table gives comparison between all the stages of Identity-based Encryption (IBE).

Table.1. Efficiency comparison for stages in IBE

	Our Scheme	IBE with Revocation
Setup	82.568 ms	83.764 ms
Key-Issuing	42.123 ms	40.369 ms
Encryption	38.142 ms	39.840 ms
Decryption	19.172 ms	21.278 ms
Key-Update	9.615 ms	10.300 ms
Proxy Re-encryption	43.412 ms	NA

VI. CONCLUSION

This paper focuses on the outsourced revocation in identity-based encryption scheme and the proxy re-encryption. We studied the outsourcing computation in IBE can be used in cloud computing, which introduced a new entity KU-CSP that computes the key-update functions and reduces the computation and storage cost at PKG. Afterwards, unrevoked users needs to periodically request on key update for time component to KU-CSP. A user can upload a file encrypted with his\her own identity on cloud and afterword can send such encrypted file to another user by re-encrypting it with receivers identity. And corresponding receiver can decrypt such ciphertext with intended private key.

VII. ACKNOWLEDGEMENT

I would like to thanks all the authors whom I have referred in this paper for giving their suggestions and making their findings and material available for us to refer.

VIII. REFERENCES

[1].W. Aiello, S. Lodha, and R. Ostrovsky, “Fast digital identity revocation,” in Advances in Cryptology (CRYPTO’98). New York, NY, USA:Springer, 1998, pp. 137-152.

[2].V. Goyal, “Certificate revocation using fine grained certificate space partitioning,” in Financial Cryptography and Data Security, S. Dietrich and R.Dhamija, Eds. Berlin, Germany: Springer, 2007, vol. 4886, pp. 247-259.

[3].F. Elwailly, C. Gentry, and Z. Ramzan, “Quasimodo: Efficient certificate validation and revocation,” in Public Key Cryptography (PKC’04), F.Bao, R. Deng, and J. Zhou, Eds. Berlin, Germany: Springer, 2004, vol. 2947, pp. 375-388.

[4].D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing,” in Advances in Cryptology (CRYPTO’01), J. Kilian, Ed. Berlin, Germany: Springer, 2001, vol. 2139. Pp. 213-229.

[5].A. Boldyreva, V. Goyal, and V. Kumar, “Identity-based encryption with efficient revocation,” in Proc. 15th ACM Conf. Comput. Commun. Security (CCS’08), 2008, pp. 417-426.

[6].A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Advances in Cryptology (EUROCRYPT’05), R. Cramer, Ed. Berlin, Germany: Springer, 2005, vol.3494, pp. 557-557.

[7].Identity-Based Encryption with outsourced Revocation in cloud Computing. J. Li, J. Li, X. Chen, C. Jia, and W. Lou. s.l. : IEEE Transaction On Computers, vol. 64, No. 2, 2015.

[8].A. Shamir, “Identity-based cryptosystems and signature schemes,” in Advances in Cryptology (CRYPTO), G. Blakley and D. Chaum, Eds. Berlin, Germany: Springer, 1985, vol.196, pp. 47-53.

[9].D. Boneh and X. Boyen, “Efficient selective-id secure identity-based encryption without random oracle,” in Advances in Cryptology (EUROCRYPT’04), C. Cachin and J. Camenisch, Eds. Berlin, Germany: Springer, 2004, vol. 3027, pp. 223-238.

[10].M. Bellare and P. Rogaway, “Random oracle are practical: A paradigm for designing efficient protocols,” in ACM Conference on Computer and Communication Security, pp. 62-73, 1993.

[11].B. Waters, “Efficient identity-based encryption without random oracles,” in Advances in Cryptology (EUROCR YPT’05), R. Cramer, Ed. Berlin, Germany: Springer, 2005, vol. 3494, pp. 114-127.

[12].E. Fujisaki and T. Okamoto, “Secure integration of asymmetric and symmetric encryption schemes,” in Advances in Cryptology (CRYPTO’99), Lecture Notes in Computer Science, vol. 1666, Springer-Verlag, pp. 537-554, 1999.