An Efficiently Subcontracted the Identity-Based Encryption for Revocation in Cloud Computing

Soujanya M A[1], Lalitha L A[2]

[1]School of Computing and Information Technology, M. Tech, Reva University, Bangalore, India

[2]School of Computing and Information Technology, Professor, Reva University, Bangalore, India

Abstract: ***Identity-Based encoding (IBE) that simplifies the general public key and certificate management at Public Key Infrastructure (PKI) is a crucial different to public key encoding. However, one amongst the most potency drawbacks of IBE is that the overhead computation at non-public Key Generator (PKG) throughout user revocation. Economical revocation has been well studied in ancient PKI setting; however the cumbersome management of certificates is exactly the burden that IBE strives to alleviate. During this paper, aiming at grappling the essential issue of identity revocation, we have a tendency to introduce outsourcing computation into IBE for the primary time and propose a revocable IBE theme within the server-aided setting. Our theme offloads most of the key generation connected operations throughout key-issuing and key-update processes to a Key Update Cloud Service supplier; exploit solely a continuing range of straightforward operations for PKG and users to perform regionally. This goal is achieved by utilizing a completely unique collusion-resistant technique: we have a tendency to use a hybrid non-public key for every user, within which Associate in Nursing AND gate is concerned to attach and sure the identity part and therefore the time part. What is more, we have a tendency to propose another construction that is obvious secure underneath the recently formulized Refereed Delegation of Computation model. Finally, we offer intensive experimental results to demonstrate the potency of our planned construction.***

Keywords: Identity-Based encoding, Public Key Infrastructure, public key, identity revocation and Cloud Service Provider.

1. Introduction

IDENTITY-BASED Encryption (IBE) is an interesting alternative to public key encryption, which is proposed to simplify key management in a certificate-based Public Key Infrastructure (PKI) by using human-intelligible identities (e.g., unique name, email address, IP address, etc) as public keys. Therefore, sender using IBE does not need to look up public key and certificate, but directly encrypts message with receiver‟s identity. Accordingly, receiver obtaining the private key associated with the corresponding identity from Private Key Generator (PKG) is able to decrypt such ciphertext. Though IBE allows an arbitrary string as the public key which is considered as appealing advantages over PKI, it demands an 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 setting, revocation mechanism is realized by appending validity periods to certificates or using involved combinations of techniques. Nevertheless, the cumbersome management of certificates is precisely the burden that IBE strives to alleviate. As far as we know, though revocation has been thoroughly studied in PKI, few revocation mechanisms are known in IBE setting. In, Boneh and Franklin suggested that users renew their private keys periodically and senders use the receivers‟ identities concatenated with current time period. But this mechanism would result in an overhead load at PKG. In another word, all the users regardless of whether their keys have been revoked or not, have to contact with PKG periodically to prove their identities and update new private keys. It requires that PKG is online and the secure channel must be maintained for all transactions, which will become a bottleneck for IBE system as the number of users grows.

In 2008, [5] presented a revocable IBE scheme. Their scheme is built on the idea of fuzzy IBE primitive [6] but utilizing binary tree data structure to record users‟ identities at leaf nodes. Therefore, key-update efficiency at PKG is able to be significantly reduced from linear to the height of such binary tree (i.e. logarithmic in the number of users).Nevertheless, we point out that though the binary tree introduction is able to achieve a relative high performance, it will result in other problems: 1) PKG has to generate a key pair for all the nodes on the path from the identity leaf node to the root node, which results in complexity logarithmic in the number of users in system for issuing a single private key. 2) The size of private key grows in logarithmic in the number of users in system, which makes it difficult in private key storage for users. 3) As the number of users in system grows, PKG has to maintain a binary tree with a large amount of nodes, which introduces another bottleneck for the global system. In tandem with the development of cloud computing, there has emerged the ability for users to buy on-demand computing from cloud-based services such as Amazon‟s EC2 and Microsoft‟s Windows Azure. Thus it desires a new working paradigm for introducing such cloud services into IBE revocation to fix the issue of efficiency and storage overhead described above.

A naive approach would be to simply hand over the PKG‟s master key to the Cloud Service Providers (CSPs). The CSPs could then simply update all the private keys by using the traditional key update technique and transmit the private keys back to unrevoked users. However, the naive approach is based on an unrealistic assumption that the CSPs are fully trusted and is allowed to access the master key for IBE system. On the contrary, in practice the public clouds are likely outside of the same trusted domain of users and are curious for users‟ individual privacy. For this reason, a challenge on how to design a secure revocable IBE scheme to reduce the overhead computation at PKG with an untrusted CSP is raised.

In this paper, we introduce outsourcing computation into IBE revocation, and formalize the security definition of outsourced revocable IBE for the first time to the best of our knowledge. We propose a scheme to offload all the key generation related operations during key-issuing and key update, leaving only a constant number of simple operations for PKG and eligible users to perform locally. In our scheme, as with the suggestion in, we realize revocation through updating the private keys of the unrevoked users. But unlike that work which trivially concatenates time period with identity for key generation/update and requires to re-issue the whole private key for unrevoked users, we propose a novel collusion-resistant key issuing technique: we employ a hybrid private key for each user, in which an AND gate is involved to connect and bound two sub-components, namely the identity component and the time component.

At first, user is able to obtain the identity component and a default time component (i.e., for current time period) from PKG as his/her private key in key-issuing. Afterwards, in order to maintain decrypt ability, unrevoked users needs to periodically request on key update for time component to a newly introduced entity named Key Update Cloud Service Provider (KU-CSP). Compared with the previous work, our scheme does not have to re-issue the whole private keys, but just need to update a lightweight component of it at a specialized entity KU-CSP. We also specify that 1) with the aid of KU-CSP, user needs not to contact with PKG in key-update, and in other words, PKG is allowed to be offline after sending the revocation list to KU-CSP. 2) No secure channel or user authentication is required during key-update between user and KU-CSP. Furthermore, we consider realizing revocable IBE with a semi-honest KU-CSP.

To achieve this goal, we present a security enhanced construction under the recently formalized Refereed Delegation of Computation (RDoC) model. Finally, we provide extensive experimental results to demonstrate the efficiency of our proposed construction.

2. Literature Survey

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, then next steps are to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

Dec. 2013.-IEEE-Privacy-assured outsourcing of image reconstruction service in cloud
Large-scale image data sets are being exponentially generated today. Along with such data explosion is the fast-growing trend to outsource the image management systems to the cloud for its abundant computing resources and benefits. How to protect the sensitive data while enabling outsourced image services, however, becomes a major concern. To address these challenges, we propose outsourced image recovery service (OIRS), a novel outsourced image recovery service architecture, which exploits different domain technologies and takes security, efficiency, and design complexity into consideration from the very beginning of the service flow and accuracy. We thoroughly analyze the privacy-protection of OIRS and conduct extensive experiments to demonstrate the system effectiveness and efficiency. For completeness, we also discuss the expected performance speedup of OIRS through hardware built-in system design.

IN 2004--"Quasimodo: Efficient certificate validation and revocation," in Public Key Cryptography
We present two new schemes for efficient certificate revocation. Our first scheme is a direct improvement on a well-known tree-based variant of the NOVOMODO system of Michal. Our second scheme is a direct improvement on a tree-based variant of a multi-certificate revocation system by Aiello, Lodha, and Ostrovsky. At the core of our schemes is a novel construct termed a QuasiModo tree, which is like a Merkle tree but contains a length-2 chain at the leaves and also directly utilizes interior nodes. This concept is of independent interest, and we believe such trees will have numerous other applications. The idea, while simple, immediately provides a strict improvement in the relevant time and communication complexities over previously published schemes.

Identity-Based Encryption

An IBE scheme which typically involves two entities, PKG and users (including sender and receiver) is consisted of the following four algorithms.

- **Setup:** The setup algorithm takes as input a security parameter and outputs the public key and the master key. Note that the master key is kept secret at PKG.
- **KeyGen:** The private key generation algorithm is run by PKG, which takes as input the master key and user"s identity. It returns a private key corresponding to the identity.
- **Encrypt**: The encryption algorithm is run by sender, which takes as input the receiver"s identity and a message to be encrypted. It outputs the cipher text.
- **Decrypt:** The decryption algorithm is run by receiver, which takes as input the cipher text and his/her private key. It returns a message or an error.

3. Problem Statement

- **KeyGen**

The key generation algorithm run by PKG takes as input–a master key , an identity , a revocation list and a time list . If, the algorithm is aborted. Otherwise, it sends the private key to user where is the identity component for private key and is its time component for current time period. Additionally, the algorithm sends an outsourcing key to

- **KU-CSP Encrypt**

The encryption algorithm run by sender takes as input–a message, an identity and a time period. It outputs the cipher text.

- **Decrypt**

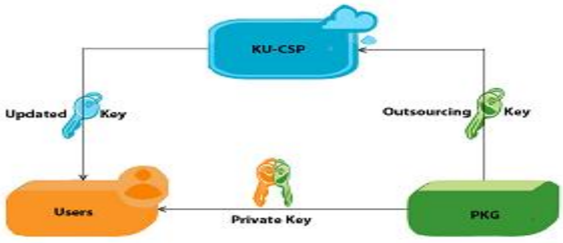
The decryption algorithm run by receiver takes as input–a cipher text encrypted under identity and time period and a private key. It outputs the original message if any, otherwise outputs. In addition, two algorithms are defined to realize revocation at KU-CSP through updating the private keys of unrevoked users.

- **Revoke**

The revocation algorithm run by PKG takes as input–a revocation list, a time list and the set of identities to be revoked. It outputs an updated time period as well as the updated revocation list and time list.

- **Key Update**

The key update algorithm run by KU-CSP takes as input–a revocation list, an identity, a time period and the outsourcing key for identity. We will show how to avoid such collusion later. Security Definition We assumes that KU-CSP in the proposed system model is semi-trusted. Specifically, it will follow our protocol but try to find out as much secret information as possible based on its possession. Therefore, two types of adversaries are to be considered as follows. Type-I adversary. It is defined as a curious user with identity but revoked before time period. Such adversary tries to obtain useful information from cipher text intended for him/her at or after (e.g. time period) through colluding with other users even if they are unrevoked. Therefore, it is allowed to ask for private key including identity component and updated time component for cooperative users. We specify that under the assumption that KU-CSP is semi-trusted, type-I adversary cannot get outsourcing key for any users. Type-II adversary. It is defined as a curious KU-CSP which aims to obtain useful information from cipher text intended for some target identity at time period. Such adversary not only possess of outsourcing keys for all users in the system, but also is able to get user‟s private key through colluding with any other user with identity . It is noted that to make such attack reasonable, we must restrict.



Compared with that for typical IBE scheme, a KU-CSP is involved to realize revocation for compromised users. Actually, the KU-CSP can be envisioned as a public cloud run by a third party to deliver basic computing capabilities to PKG as standardized services over the network. Typically, KU-CSP is hosted away from either users or PKG, but provides a way to reduce PKG computation and storage cost by providing a flexible, even temporary extension to infrastructure. When revocation is triggered, instead of re-requesting private keys from PKG in [4], unrevoked users have to ask the KU-CSP for updating a lightweight component of their private keys. Though many details are involved in KU-CSP‟s deployment, in this paper we just logically envision it as a computing service provider, and concern how to design secure scheme with an untrust KU-CSP.

Based on the system model proposed, we are able to define the outsourced revocable IBE scheme. Compared with the traditional IBE definition, the KeyGen Encrypt and Decrypt algorithms are redefined as follows to integrate time component. Note that two lists and are utilized in our definition, where records the identities of revoked users and is a linked list for past and current time period.

KeyGen: The key generation algorithm run by PKG takes as input–a master key , an identity , a revocation list and a time list. If, the algorithm is aborted. Otherwise, it sends the private key to user where is the identity component for private key and is its time component for current time period. Additionally, the algorithm sends an outsourcing key to KU-CSP.

Encrypt: The encryption algorithm run by sender takes as input–a message, an identity and a time period. It outputs the cipher text.

Decrypt: The decryption algorithm run by receiver takes as input–a cipher text encrypted under identity and time period and a private key. It outputs the original message if any, otherwise outputs. In addition, two algorithms are defined to realize revocation at KU-CSP through updating the private keys of unrevoked users.

Revoke: The revocation algorithm run by PKG takes as input–a revocation list, a time list and the set of identities to be revoked. It outputs an updated time period as well as the updated revocation list and time list.

Key Update: The key update algorithm run by KU-CSP takes as input–a revocation list , an identity , a time period and the outsourcing key for identity . It outputs user‟s updated time component in private key if his identity does not belong to , otherwise, outputs .

In this paper, we discuss user revocation that is how to deprive users of decrypt ability even if they have been issued their private keys. To this end, we embed a time period into private key in a clever manner for revocation. Specifically, in the same example illustrated in Section 2.2, Alice in our setting not only encrypts message with Bob‟s email address “bob@company.com” but also with current time period (e.g., “Thu Jul 18 2013”). When receives the encrypted email, Bob then obtains his private key consisting of an identity component and a time period component from PKG. With the both appropriate components, the email can be read.

Suppose Bob is compromised. Then, the time components of all the other users are updated by KU-CSP with a new time period (e.g., “Fri Jul 19 2013”). From then on, the message sent to Bob should be encrypted with Bob‟s email address and the updated time period. Since Bob does not have the time component corresponding to the updated time period, the following encrypted messages cannot be decrypted by Bob even if they are intended for him.

The challenge in designing the outsourced revocable IBE scheme is how to prevent collusion between Bob and other unrevoked dishonest users. Specifically, a dishonest user (named Eve) can share her updated time component (i.e., “Fri Jul 19 2013”) with Bob, and help Bob decrypt cipher text even if Bob just has the previous one (i.e., “Thu Jul 18 2013”). We will show how to avoid such collusion later.

4. Efficient IBE with Outsourced Revocation

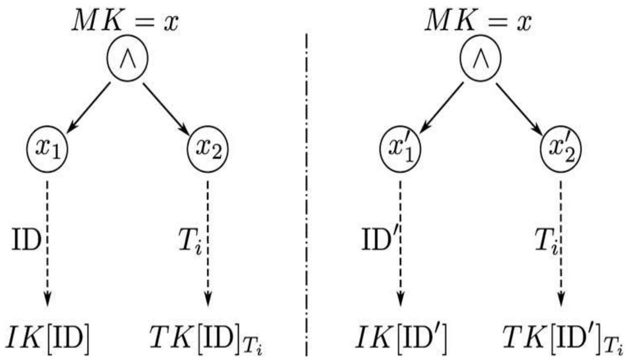
Intuition

In order to achieve efficient revocation, we introduce the idea of “partial private key update” into the proposed construction, which operates on two sides: 1) we utilize a “hybrid private key” for each user in our system, which employs an AND gate connecting two sub-components namely the identity component and the time component respectively. is generated by PKG in key-issuing but is updated by the newly introduced KU-CSP in key update; 2) In encryption, we take as input user‟s identity as well as the time period to restrict decryption, more precisely, a user is allowed to perform successful decryption if and only if the identity and time period embedded in his/her private key are identical to that associated with the cipher text. Using such skill, we are able to revoke user‟s decrypt ability through updating the time component for private key by KU-CSP.

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 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 to update the time component for identity. Suppose a user with identity is revoked at. Even if he/she is able to obtain for identity, the revoked user still cannot decrypt cipher text encrypted under.

4.1 Proposed Construction

An identity-based encryption with outsourced revocation scheme is semantically secure against adaptive chosen-cipher text attack (IND-ID-CCA) if no polynomials bounded adversary has a non-negligible advantage against challenger in security game for both type-I and type-II adversary. Finally, beyond the CCA security, we also specify that 1) An IBE with outsourced revocation scheme is INDID-CPA secure (or semantically secure against chosen plaintext attack) if no polynomial time adversary has non-negligible advantage in modified games for both type-I and type-II adversary, in which the decryption oracle in both phase 1 and phase 2 is removed; 2) An IBE with outsourced revocation scheme is secure in selective model if no polynomial time adversary has non-negligible advantage in modified games for both type-I and type-II adversary, in which the challenge identity and time period is submitted before setup.



4.1: A comparison on generating private key for two different users.

Finally, we emphasize that the idea behind our construction is to realize revocation through updating the time component in private key. Therefore, the key point is to prevent revoked user from colluding with other users to re-construct his/her private key. As declaring in intuition, such collusion attack is resistant in our proposed construction due to the random split on for each user. Specifically, as shown in Fig. 4.1 in which is an AND gate connecting two sub-components, if two different users call for their private keys, PKG will obtain two randomly splits ( ) and ( ) with the complementary that and . and are used to produce the identity component for and respectively, while the time component is separately generated from and . By the reason that the complementary exists between and as well as and , the identity component and time component should accordingly have a “verification” in private key. With such “verification”, even if a curious user obtains time component of other users, he/she cannot forge a valid private key for himself to perform decryption successfully.

4.2 Key Service Procedures

Based on our algorithm construction, as shown in Fig. 4.2, the key service procedures including key-issuing, key-update and revocation in proposed IBE scheme with outsourced revocation work as follows. Key-issuing. We require that PKG maintains a revocation list and a time list locally. Upon receiving a private key request on, PKG.

KeyGen: to obtain private key and outsourcing key. Finally, it sends to user and ( ) to KUCSP respectively. As described in intuition, for each entry ( ) sent from PKG, KU-CSP should add it into a locally maintained user list.

Key-update: If some users have been revoked at time period, each unrevoked user needs to send key-update request to KU-CSP to maintain decrypt ability. Upon receiving the request on identity, KU-CSP runs Key Update to obtain. Finally, it sends such time component back to user who is able to update his/her private key.

Revocation: Similar to key-update, if a revoked user sends a key-update request on identity , KU-CSP.

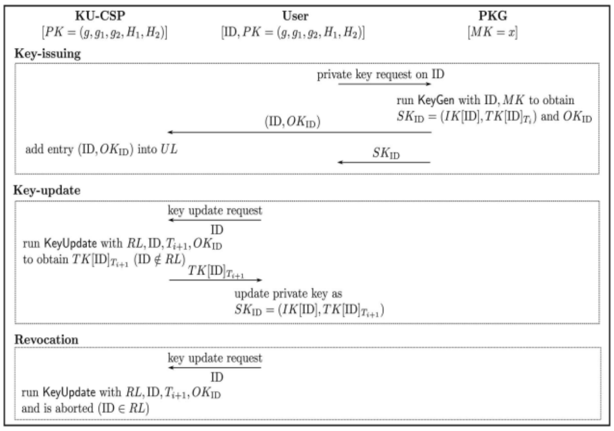

KU-CSP
$[PK = (g, g_1, g_2, H_1, H_2)]$
User
$[\mathrm{ID}, PK = (g, g_1, g_2, H_1, H_2)]$
PKG
$[MK = x]$
Key-issuing
private key request on ID
run KeyGen with ID, MK to obtain
$SK_{\mathrm{ID}} = (IK[\mathrm{ID}], TK[\mathrm{ID}]_{T_i})$ and OK_{ID}
$(\mathrm{ID}, OK_{\mathrm{ID}})$
add entry $(\mathrm{ID}, OK_{\mathrm{ID}})$ into UL
SK_{ID}
Key-update
key update request
ID
run KeyUpdate with $RL, \mathrm{ID}, T_{i+1}, OK_{\mathrm{ID}}$
to obtain $TK[\mathrm{ID}]_{T_{i+1}}$ $(\mathrm{ID} \notin RL)$
$TK[\mathrm{ID}]_{T_{i+1}}$
update private key as
$SK_{\mathrm{ID}} = (IK[\mathrm{ID}], TK[\mathrm{ID}]_{T_{i+1}})$
Revocation
key update request
ID
run KeyUpdate with $RL, \mathrm{ID}, T_{i+1}, OK_{\mathrm{ID}}$
and is aborted $(\mathrm{ID} \in RL)$

Figure 4.2: Protocol for key-issuing, key update and revocation.

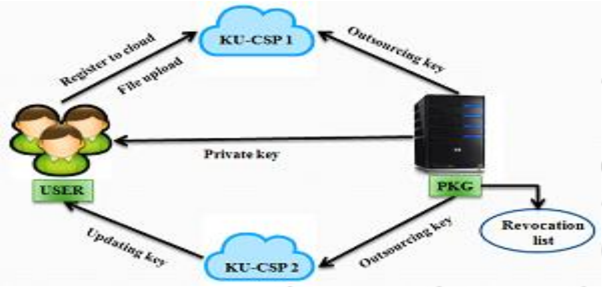


Figure 2.2: System model with two KU-CSPs.

5. Advanced Construction Under Refereed Delegation of Computation Model

We will attempt to propose a security enhanced construction under the under the recently formalized RDoC model.

5.1 Advanced Construction

RDoC model originates from the model of refereed games in, and is later formalized. In RDoC model, the client is able to interact with multiple servers and it has a right output as long as there exists one server that follows the proposed protocol. One of the most advantages of RDoC over traditional model with single server is that the security risk on the single server is reduced to multiple servers involved in. As the result of both the practicality and utility, RDoC model recently has been widely utilized in the literature of outsourced computation.

In order to apply RDoC to our setting, we introduce other independent KU-CSPs. For simplicity, in the rest of paper, weonly focuse on the case that as shown in Fig. 5. Furthermore, we have three requirements in such model: 1) At least one of the KU-CSPs is honest. 2) Computational complexity at the honest KU-CSP is not much more than the other required performing revocation. 3) PKG"s running time would be much smaller than required to directly perform revocation.

We figure out that the challenge to realize such advanced construction is to demand that and cannot be leaked at the same time. To achieve this goal, we randomly split into and which will be separately used by the two KU-CSPs to produce partial time component and. After receiving the two partial time components, user performs a production

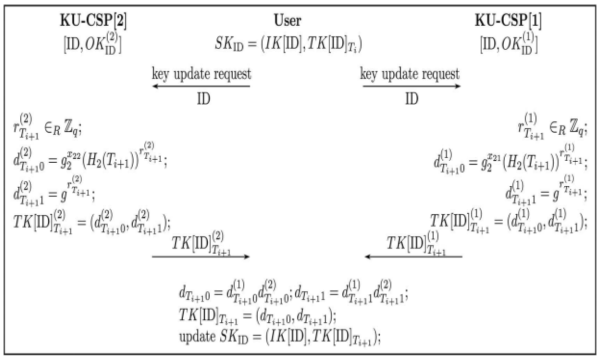

KU-CSP[2]
$[\mathrm{ID}, OK_{\mathrm{ID}}^{(2)}]$
User
$SK_{\mathrm{ID}} = (IK[\mathrm{ID}], TK[\mathrm{ID}]_{T_i})$
KU-CSP[1]
$[\mathrm{ID}, OK_{\mathrm{ID}}^{(1)}]$
key update request
ID
key update request
ID
$r_{T_{i+1}}^{(2)} \in_R \mathbb{Z}_q;$
$d_{T_{i+1}0}^{(2)} = g_2^{x_{22}} (H_2(T_{i+1}))^{r_{T_{i+1}}^{(2)}};$
$d_{T_{i+1}1}^{(2)} = g^{r_{T_{i+1}}^{(2)}};$
$TK[\mathrm{ID}]_{T_{i+1}}^{(2)} = (d_{T_{i+1}0}^{(2)}, d_{T_{i+1}1}^{(2)});$
$r_{T_{i+1}}^{(1)} \in_R \mathbb{Z}_q;$
$d_{T_{i+1}0}^{(1)} = g_2^{x_{21}} (H_2(T_{i+1}))^{r_{T_{i+1}}^{(1)}};$
$d_{T_{i+1}1}^{(1)} = g^{r_{T_{i+1}}^{(1)}};$
$TK[\mathrm{ID}]_{T_{i+1}}^{(1)} = (d_{T_{i+1}0}^{(1)}, d_{T_{i+1}1}^{(1)});$
$TK[\mathrm{ID}]_{T_{i+1}}^{(2)}$
$TK[\mathrm{ID}]_{T_{i+1}}^{(1)}$
$d_{T_{i+1}0} = d_{T_{i+1}0}^{(1)} d_{T_{i+1}0}^{(2)}; d_{T_{i+1}1} = d_{T_{i+1}1}^{(1)} d_{T_{i+1}1}^{(2)};$
$TK[\mathrm{ID}]_{T_{i+1}} = (d_{T_{i+1}0}, d_{T_{i+1}1});$
update $SK_{\mathrm{ID}} = (IK[\mathrm{ID}], TK[\mathrm{ID}]_{T_{i+1}});$

Figure 5.1: Key Update and Key Combine in advanced construction

5.2 Security Analysis

As a stronger adversary model, RDoC captures much more meaning beyond the "honest-but-curious" sense that is curious user is allowed to cooperate with at most servers if servers are involved. To accommodate to this case, we modify the private key oracle slightly to adapt to a pair of outsourcing keys and introduce another outsourcing key extraction oracle for Type-I adversary as follows. It is noted that the challenger is required to maintain an empty set to restrict adversary accessing the whole outsourcing key for some identity. This coincides with the assumption that at least one of the KU-CSPs is honest.

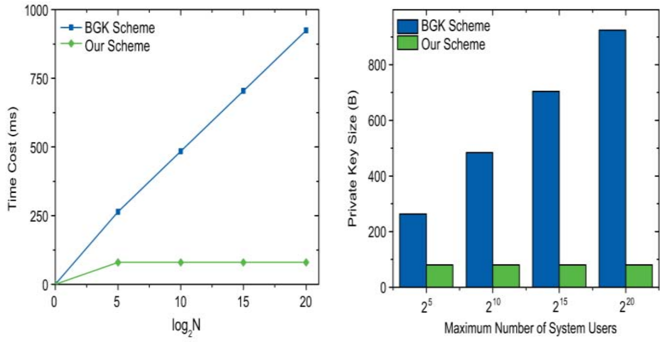
Private key extraction oracle

Upon receiving private key request on, challenger runs to obtain the private key and a pair of outsourcing keys. After adding the entry ( ) into, return. Outsourcing key extraction oracle. Upon receiving the partial outsourcing key request on to the KU-CSP, challenger firstly checks weather. If so the oracle is aborted. Otherwise, if there exists an entry ( ) in, after setting return.

Table 1: Efficiency Comparison for Stages in Revocable IBE

	Our Scheme	IBE without Revocation [4]
Setup	83.764 ms	80.233 ms
Key-Issuing	40.369 ms	20.121 ms
Encryption	39.840 ms	24.595 ms
Decryption	21.278 ms	10.285 ms
Key-Update	10.300 ms[1]	—-

[1] This time cost is evaluated at KU-CSP.



(a) Responding Time for Single Key Generation Request

(b) Private Key Size

Figure 5.2: Comparisons in key-issuing (is the maximum number of users in the system)

6. Performance Evaluation

We will provide a thorough experimental evaluation of the construction proposed. We build our tested by using 64-bit M2 high-memory quadruple extra large Linux servers in Amazon EC2 platform as KU-CSP, and a Linux machine with Intel(R) Core(TM)2 Duo CPU clocked at 2.40 GHz and 2 GB of system memory as the user and PKG. Note that in all the evaluations, the groups G and G are selected in 160-bit and 512-bit length respectively.

6.1 Performance Evaluation for Overall Scheme

Firstly, we aim to evaluate the efficiency of our outsourced revocable scheme by comparing the total time taken during each stage with the original IBE which does not consider revocation.

In Table 1, we examine the time cost of executing individual stage by the both schemes. It is not surprising to see that our scheme takes more time because we consider the revocability issue. Note that our scheme shares the same setup algorithm with the IBE scheme in. Our key-issuing stage is relative longer than that in the IBE scheme. This is because we embed a time component into each user‟s private key to allow periodically update for revocation, resulting that some additional computations2 are needed in our scheme to initialize this component. Our encryption and decryption is slightly longer than the IBE scheme, which is also due to the existence of the time component. The user needs to perform an additional encryption/decryption for this component, rather than just encrypt/decrypt the identity component.

6.2 Performance Evaluation for Revocation

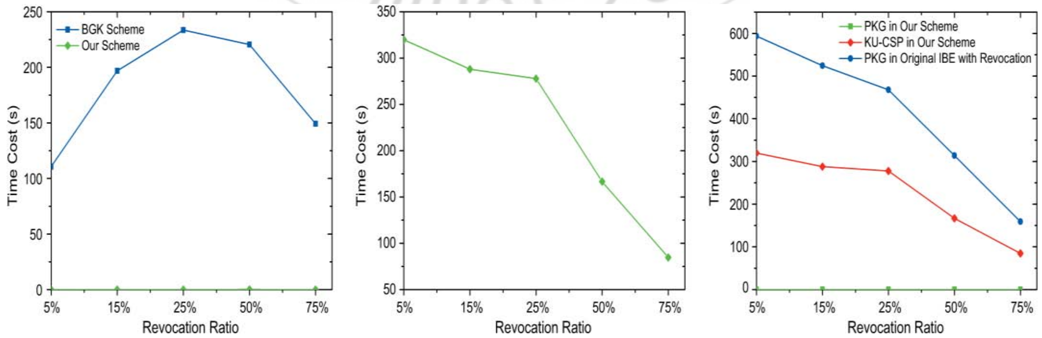
Secondly, we attempt to simulate the scenario of multi-user revocation, and show an extensive comparison between our outsourced revocation scheme and another revocable IBE scheme–BGK scheme. Note that in this set of experiments, we use a 32-bit integer to identify each node in binary tree which is utilized in BGK scheme for managing users.

6.2.1 Key-Issuing Stage

In Fig. 5(a), we vary the maximum number of users in the system and show the responding time for a single key generation request. It is not hard to see that the responding time in BGK scheme is in proportion of where is the Maximum number of users in system. This is because a binary tree is utilized to manage all the users, each leaf node of which is assigned to a single user in system. During key-issuing, PKG has to perform computation on all the nodes in the path from the corresponding leaf node to root node.

6.2.2 Key Update Stage

In this experiment, we randomly pick 5% to 75% users and compare the total time of updating private keys for the rest users. For simplicity, we just illustrate an example and compare the key-update time at PKG in revocation in the case of system users in Fig. 5(a). It can be seen that the efficiency curve of BGK scheme shows a parabolic shape, and at the 25% revocation ratio, the efficiency achieves the lowest point in our evaluation. This is because it is the gap that the leaf nodes to be revoked have a large number but low aggregation degree, which requires that we have to update a lot of internal nodes for key-update.



(a) Key Update at PKG

(b) Key Update at KU-CSP

(c) Key Update Comparison with [4]

Figure 5: Comparisons in key update

7. Related Work

7.1 Revocable IBE

Introduced by and firstly implemented by Boneh and Franklin [4] as well as [14], IBE has been researched intensively in cryptographic community.

On the aspect of construction, these first schemes were proven secure in random oracle. Some subsequent systems achieved provable secure in standard model under selective-ID security, or adaptive-ID security. Recently, there have been multiple lattice-based constructions for IBE systems.

Nevertheless, concerning on revocable IBE, there is little work presented. As mentioned before, Boneh and Franklin‟s suggestion is more a viable solution but impractical. Hanaoka et al. Proposed a way for users to periodically renew their private keys without interacting with PKG. However, the assumption required in their work is that each user needs to possess a tamper-resistant hardware device. Another solution is mediator-aided revocation [24], [25]: In this setting there is a special semi-trusted third party called a mediator who helps users to decrypt each cipher text. If an identity is revoked then the mediator is instructed to stop helping the user. Obviously, it is impractical since all users are unable to decrypt on their own and they need to communicate with mediator for each decryption. Recently, Lin et al. proposed a space efficient revocable IBE mechanism from non-monotonic Attribute-Based Encryption (ABE), but their construction requires times bilinear pairing operations for a single decryption where the number of revoked users is.

As far as we know, the revocable IBE scheme presented by Boldyreva et al. Remains the most effective solution right now. Libert and Vergnaud improved Boldyreva‟s construction to achieve adaptive-ID security. Their work focused on security enhanced, but inherits the similar disadvantage as Boldyreva‟s original construction. As we mentioned before, they are short in storage for both private key at user and binary tree structure at PKG.

8. Conclusion

In this paper, focusing on the critical issue of identity revocation, we introduce outsourcing computation into IBE and propose a revocable scheme in which the revocation operations are delegated to CSP. With the aid of KU-CSP, the proposed scheme is full-featured: 1) It achieves constant efficiency for both computation at PKG and private key size at user; 2) User needs not to contact with PKG during key update, in other words, PKG is allowed to be offline after sending the revocation list to KU-CSP; 3) No secure channel or user authentication is required during key-update between user and KU-CSP. Furthermore, we consider realizing revocable IBE under a stronger adversary model. We present an advanced construction and show it is secure under Do model, in which at least one of the KU-CSPs is assumed to be honest. Therefore, even if a revoked user and either of the KU-CSPs collude, it is unable to help such user re-obtain his/her decrypt ability. Finally, we provide extensive experimental results to demonstrate the efficiency of our proposed construction.

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. 15thACMConf. 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] R. Canetti, B. Riva, and G. N. Rothblum, “Two 1-round protocols for delegation of computation,” Cryptology ePrint Archive, Rep. 2011/ 518, 2011 [online]. Available: http://eprint.iacr.org/2011/518.

[8] U. Feige and J. Kilian, “Making games short (extended abstract),” in Proc. 29th Annu. ACM Symp. Theory Comput. (STOC‟97), 1997, pp. 506–516.

[9] S. Hohenberger and A. Lysyanskaya, “How to securely outsource cryptographic computations,” in Proc. 2nd Int. Conf. Theory Cryptography (TCC‟05), 2005, pp. 264–282.

[10] R. Canetti, B. Riva, and G. Rothblum, “Two protocols for delegation of computation,” in Information Theoretic Security, A. Smith, Ed. Berlin, Germany: Springer, 2012, vol. 7412, pp. 37–61.

[11] X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, “New and secure outsourcing algorithms of modular exponentiations,” in Proc. 17th Eur. Symp. Res. Comput. Security (ESORICS), 2012, pp. 541–556.

[12] M. J. Atallah and K. B. Frikken, “Securely outsourcing linear algebra computations,” in Proc. 5th ACM Symp. Inf. Comput. Commun. Security (ASIACCS‟10), 2010, pp. 48–59.

[13] 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.