Efficient Trust Based Information Sharing Schemes over Distributed Collaborative Networks

Huang Lin, Xiaoyan Zhu, Chi Zhang, Yuguang Fang and Zhenfu Cao

***Abstract*—In distributed collaborative networks such as peer-to-peer systems and wireless ad hoc networks, secure communications, information sharing and dissemination heavily depend on effective trust management. Recently, trust based encryption (TBE) has been proposed to deal with secure information sharing and dissemination for such networks. Unfortunately, the previous schemes proposed are not efficient in terms of communications overhead. In this paper, we provide a trust based encryption schemes which can significantly improve the communication overhead. The proposed scheme is based on a recently proposed revocable identity based encryption technique which can reduce the communication overhead at the central trust authority from the linear order of the number of the users in the network to the linear order of the number of the revoked users. Besides, the communication overhead at each receiver is also significantly reduced.**

***Index Terms*—Trust management, Secure information sharing, Trust based encryption, Revocable IBE**

I. Introduction

Reputation systems have served as an important tool in establishing trust in the distributed networks. Users in such networks can offer their reputation rating for a certain network node, service or product based their interactions or use experience. They can also derive evidence from other nodes' ratings or feedback and come up with their own rating about certain service or the trust towards a node and/or the service it provides. Most of the current web message board systems [**?**] also employ users' feedback rating on published news to help highlight the quality of the contents. In the recently emerging distributed networks, such as peer-to-peer (P2P) networks, various rating systems based on reputation are designed to achieve different security goals, such as preventing Sybil Attacks [**?**] or establishing social trust [**?**]. In all these reputation based systems, each user or entity is evaluated by its reputation value and treated differently according to this value. This shares some similarities to the real world scenario in which people tend to have certain trust evaluation on the others they contact with and react differently (although we might not explicitly give each other a reputation value, we do have some psychological value in our minds). For example, people are inclined to trust those who they are familiar with more than the strangers.

Reputation system has also been applied in access control for both traditional networks and mobile communication systems. For instance, in the recently booming online social networks (OSNs), such as Twitter and Facebook, people tend to share various personal profile information with friends or even strangers they have met online. These information could be divided into different security levels based on its privacy level. Apparently, user's reputation or trust levels between different users gathered from users' interactions or feedback from other users' interaction experience could serve as a base to realize secure information sharing and dissemination.

Thus, given a reputation system in place for a distributed network, how to efficiently enable secure information sharing and dissemination based on the reputation rating is an interesting and challenging design question. For example, a user may want to share his information with reputation level higher than certain threshold while hiding his information from those with rating lower than the threshold, how can this be done efficiently? This has many potential interesting applications. A server can only allow its multicast/broadcast service to be accessed by those with good reputation during the service delivery. A user in an OSN may only allow users with good reputation to access its profile or blogs. Trust based encryption (TBE) technique has been proposed to address these kinds of issues [1]. In this proposed TBE scheme, each user is evaluated with a reputation rating value, say, r, where $r \in [0, 1)$ is a rational number equivalent to $r = a/2^{\kappa}, a \in [0, 2^{\kappa})$ and $u = 2^{\kappa}$ represents the granularity of reputation rating. The lower the user rating value is, the more trustworthy this user is. There is a trusted authority (TA) deployed to be responsible for distributing a private key for each user according to its identity and rating value. The basic idea runs as follows. A sender, say Bob, wants to communicate with a receiver, say, $Alice$. Bob encrypts the message using $Alice$'s identity, a reputation requirement $[0, R]$ and the current communication round t. $Alice$ can successfully decrypt the encrypted message only when her rating value r in the current communication round falls into the range requirement $[0, R]$. This indeed solves the problem we raised for information sharing.

Several variations of such TBE schemes, under either the symmetric key or public key framework, were also provided in [1] to address various problems. However, as also pointed out in [1], the symmetric key framework requires TA to be online for distributing both the sender and receiver fresh private key whenever they want to communicate with each other, which creates significant communication load to the system, which may not be practical. On the other hand, all three proposed public key TBE schemes [1] are based on identity based encryption (IBE) technique. The first scheme based on basic IBE techniques achieves a $\mathcal{O}(\log(u))$=$\mathcal{O}(\kappa)$ size private key and $\mathcal{O}(\kappa)$ size ciphertext. The receiver is required to get a fresh private key based on the current reputation rating value from TA each communication round. Therefore, the communication workload at TA should be of size $\mathcal{O}(nT\kappa)$, where n is the number of the system users and T is the maximum number of communication rounds between any two communication parties. The second scheme, which is based on the ID-based multi receiver key encapsulation mechanism (ID-MR-KEM), has a similar performance compared with the first scheme. The third scheme is based on the hierarchical IBE scheme and achieves a constant size ciphertext and a much larger private key size, which might be less favorable for most applications. Both the second and third schemes require the receiver to get its updated private key at each communication round.

As we can observe, the proposed TBE schemes tend to generate too much communication overhead (traffic load) to the TA. This paper is to improve the network efficiency by developing more efficient TBE schemes. We provide a TBE scheme which can significantly reduce the workload at TA and receivers. The proposed TBE scheme is based on the current revocable IBE system [2]. The underlying idea of this scheme is to let the TA periodically publish update information to revoke those whose reputation scores have significantly changed in a prefixed time period. The published update information size is dependent on the number of revoked users. Thus, instead of getting her private key from the TA each communication round, the receiver only needs to refresh her private key from the TA when her reputation score changes. Therefore, the traffic burden for the TA to distribute the private keys will then depend on the number of the revoked users.

The rest of the paper is organized as follows. We start with a brief introduction to the basic idea of the original TBE scheme in the next section. After then, we present a TBE scheme constructed from the revocable IBE scheme. Finally, we conclude this paper with some future work.

II. A brief introduction to original TBE scheme

To better understand our TBE scheme, we first present the TBE scheme proposed in [1]. Consider a scenario where the sender Bob specifies a trust rating (or reputation score) R when encrypting a message for the receiver $Alice$. The scheme is to guarantee that the decryption is successful only when $Alice$ owns a secret key whose trust rating value r satisfies $r \leq R$. The secret keys are required to depend on temporal information, identities and trust rating, so that keys for one communication round cannot be used for the next round. It is assumed that there is a secure channel between the TA and each user to guarantee the secure delivery of user's secret key.

The identity based encryption (IBE) system ([3], [4]) serves as an important tool for the proposed TBE scheme. There are three parties in the IBE scheme: the Private Key Generator (PKG), which is a trusted authority holding a master key mk and responsible for initializing the system, publishing the system parameter pk and outputting a private key sk_{id} for each system user with identification (ID) id by running an extraction algorithm that takes mk and id as input; the encryptor, who runs an encryption algorithm taking as input the message M, the receiver identity id and the public key pk to generate a ciphertext $C_{id}(M)$. The receiver id' will input the received ciphertext $C_{id}(M)$ and its private key $sk_{id'}$ to the decryption algorithm. The algorithm will output the original message M if $id = id'$ and $\perp$ otherwise.

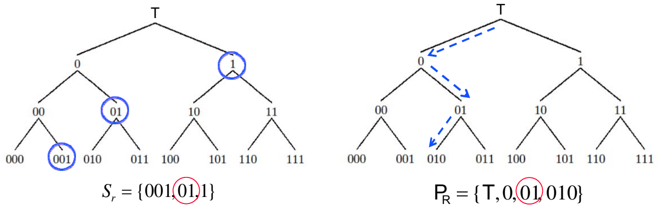

Fig. 1. Basic idea of the TBE in [1]: ***rating set*** r=$\frac{1}{8}$, ***range set*** for $[0, R]$=$[0, \frac{1}{4}]$, u=8

The identity based TBE scheme requires an online TA, from whom the receiver can obtain its private key corresponding to trust rating value r in the communication round t. The binary tree-based technique for range queries over the encrypted data [5] is used to generate the private keys with the desired property. As illustrated in [1], the root of the binary tree with depth d is labeled $\top$ (representing the string of length 0), a left-child at node s will be labeled as $s0$, and a right-child node will be labeled as $s1$ as shown in Fig.1. As a result, the leaves will be labeled from left-to-right by d-bit strings, beginning $0 \cdots 0$ and ending $1 \cdots 1$. Each binary string $b_0 \cdots b_{d-1}$ will be uniquely associated with a real number $r = \sum_{i=0}^{\ell-1} b_i 2^{-(i+1)}$ in the interval $[0, 1)$. In the identity based TBE [1], the user with trust value r of form a/u is associated with an identity set S_r covering the interval $[a, u)$, i.e., a minimal set of subtrees that cover the leaf nodes for the range a to u. This identity set is denoted as ***rating set*** in the context. In order to generate a ciphertext for a range $[0, R]$ (corresponding to the range requirement R), the sender will first find out all the nodes on the path from the root to the leaf node R, under which the message will be encrypted. All these identity nodes on the path form the ***range set***. For instance, in Fig. 1 with $r = 1/8$ and $R = 1/4$, the rating set is $\{1, 01, 001\}$ and the range set is $\{\top, 0, 01, 010\}$. Thus, a message M will be encrypted under the identity set $\{id||\top||t, id||0||t, id||01||t, id||010||t\}$, where id is the recipient identity, the respective trust range is $[0, \frac{1}{4}]$, and t denotes the communication round. In other words, the corresponding ciphertext $C_{id,[0,R],t}(M)$ consists of ciphertext $C_{id||\top||t}(M)$, $C_{id||0||t}(M)$, $C_{id||01||t}(M)$ and $C_{id||010||t}(M)$. The private key for a recipient id with the trust rating value $r = \frac{1}{8}$ is assigned with the identity set $\{id||1||t, id||01||t, id||001||t\}$ with the corresponding rating set $\frac{1}{8}$ is $\{1, 01, 001\}$. In other words, this recipient will have a private key consisting of $sk_{id||1||t}$, $sk_{id||01||t}$ and $sk_{id||001||t}$. The decryption is successful due to the intersected identity $id||01||t$ between the ***rating set*** and ***range set***. The recipient could simply run the decryption algorithm on the ciphertext $C_{id||01||t}(M)$ using its private key $sk_{id||01||t}$ to obtain the message M.

III. System Model and Design Goals

The system model for our improved TBE schemes is close to that of the original TBE system[1]. We consider a scenario where a user shares private information with the other users in a distributed network. The sender and the receiver might be familiar with each other or the receiver might be a stranger to the sender. In the first case, the receiver's identity is naturally known to the sender. In the other case, the receiver would let her identity known to the sender when she asks for information sharing. Each user in the system should have a reputation rating value r in the similar form as in the original TBE scheme, which is assigned by the TA according to a ceratin reputation system. We do not put much emphasis on the design of the underlying rating system as in the original TBE scheme and we assume there already exists a reputation rating system which could provide a fair rating value for each user and objectively reflect the behavior of this user. The TA, equipped with the reputation system, distributes secret keys for users according to their identities and rating values securely (we assume there is a secure channel for online private key distribution). Similar to the original TBE system, a sender encrypts a message under a reputation range requirement $[0, R]$ and the receiver's identity in both of our two proposed TBE systems. The receiver could successfully decrypt the corresponding ciphertext only when the rating value r falls into this range $[0, R]$.

We focus on improving the efficiency of the TBE systems, which is measured by the communication overhead and memory storage cost. The communication overhead is mainly determined by the size of ciphertext delivered from a sender to a receiver and the communication load between a system user and TA. The communication load between a system user and TA depends on the size of the private keys from TA to the users and the communication rounds between them. The communication load should also include the workload of a system user for processing those private keys after receiving them. Our scheme improves the communication overhead by reducing the number of the communication rounds (i.e., reducing the signaling traffic cost for private key updates).

IV. TBE scheme from R-IBE scheme

In this section, we demonstrate how the revocable identity based encryption (R-IBE) [2] could be used to improve the communication overhead of our TBE system by reducing the communication traffic between users and the TA.

In what follows, we first provide a brief introduction to the R-IBE system. We then show how the application of R-IBE to design our TBE system can significantly reduce the communication rounds between users and the TA.

A. A brief introduction to the revocable IBE

Revocable IBE scheme deals with the identity revocation in the IBE system. Since user's private key might be stolen or expired, the revocable IBE provides a mechanism to prevent these "corrupted" private keys from being used to decrypt ciphertext. The system life time of a R-IBE scheme can be divided into time periods. At the beginning of the system operation, each user obtains a private key for its identity from the TA. A message is encrypted under the receiver's identity id and the current time period t. In order to successfully decrypt the ciphertext, the receiver id should manage to generate the decryption key for the current time period t. Notice that the decryption key and the private key are different in a R-IBE system. Those who owns a private key might not be able to generate a decryption key for a certain time period if his identity is revoked in this period. At the beginning of each time period, the TA publishes the update information which only allows the unrevoked users to update their private key in order to generate the decryption key for the current time period. In this way, those revoked users (or identities) will be deprived of their decryption ability.

The R-IBE scheme usually consists of the following seven algorithms:

1) **R-Setup**$(1^{\lambda}, n)$: This algorithm is run by the TA, which takes as input a security parameter and the number of system users n. After running this algorithm, the TA will publish the public key pk. This algorithm will also output a master key msk and a initially empty revocation list rl for the TA.
2) **R-PriKeyGen**$(msk, \textbf{id})$: This algorithm is also run by the TA, and TA will input an arbitrary identity string **id** and the master key msk, and outputs the user private key $sk_{\textbf{id}}$ for that identity after running this algorithm.
3) **R-KeyUpdate**(pk, msk, t, rl): The TA executes this key update algorithm to publish the update information for the time period t. The TA inputs the system parameters including the public key pk, the master key msk, the key update time t and the revocation list rl, and then outputs the key update information ku_t. Here, the revocation list rl specifies the revoked user identity **id** and other related information. Although the key update information ku_t is publicly accessible, they are useless for those revoked identities.
4) **R-DecryKeyGen**$(sk_{\textbf{id}}, ku_t)$: This decryption key generation algorithm is run by the unrevoked user each time after the TA publishes the update information ku_t. The unrevoked users runs this algorithm by taking as the input the user private key $sk_{\textbf{id}}$ and the key update information ku_t, and then outputs the decryption key $dk_{\textbf{id},t}$. It outputs $\perp$ if a revoked user tries to run this algorithm.
5) **R-Enc**$(pk, \textbf{id}, t, M)$: The encryption algorithm is run by a sender. It takes as the input the public key pk, the receiver identity **id**, the current time period t and the message M, and then outputs the ciphertext $C_{\textbf{id},t}(M)$.
6) **R-Dec**$(dk_{\textbf{id},t}, C_{\textbf{id},t}(M))$: The receiver runs this decryption algorithm by inputting the decryption key $dk_{\textbf{id},t}$ and the ciphertext $C_{\textbf{id},t}(M)$, and then outputs a message M or a special symbol $\perp$. The consistency condition for the decryption is: if the receiver identity **id** is unrevoked at the time period t, then the decryption algorithm will output the message M if the identity is unrevoked, and $\perp$ otherwise because those revoked users cannot generate the respective decryption key $dk_{\textbf{id},t}$.
7) **R-Revocation**: The revocation algorithm is run by the TA, which takes as the input the identity to be revoked **id** and the revocation list rl, and then outputs an updated revocation list rl.

The scheme [2] proposed by Boldyreva et al adopts the binary tree structure and the fuzzy identity based encryption

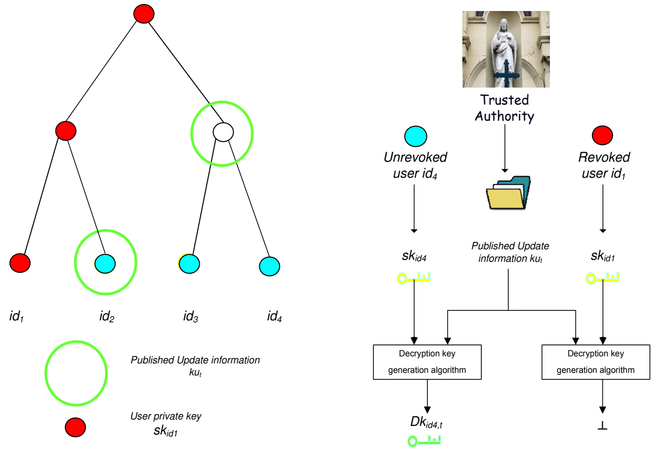

Fig. 2. Basic idea of a R-IBE: each unrevoked user can use the update information for the time period t to generate the respective decryption key while the revoked users cannot

as the underlying scheme. The basic idea of a R-IBE scheme (see Fig. 2) is that: for a system with n users, the PKG will first generate a binary tree with at least n leaf nodes. Each user will correspond to one unique leaf node. The user private key is assigned according to a node set consisting of the nodes on the path from the root to its own leaf node. The periodically published update information is generated according to the minimal node set which only covers those unrevoked users. That is exactly why the update information will be useless to those revoked users. Check the left sub-figure in Fig.2, the key update information ku_t is generated according to the big green circle nodes which only cover the path nodes contained in the private key node set for the unrevoked users, i.e. $\{id_2, id_3, id_4\}$. It is also easy to observe that ku_t does not cover any of the path nodes in the private key node set for id_1, and thus will be useless to id_1. The private key size is $\mathcal{O}(\log n)$ and the size of the key pdate information is $\mathcal{O}(v \log(n/v))$, where v is the number of the revoked users. After running the key update algorithm, the decryption key for each user is constant size.

B. Improved TBE scheme

As the reputation score (trust rating value) of each user is fluctuating with time, or nodes, so their trust based private keys, are subject to compromise, our TBE system should provide a mechanism to revoke a user's reputation key when his/her reputation changes. Although a game theoretic mechanism [1] was designed to ensure a rational user to honestly report his/her current estimated rating value to the TA, this still cannot deter those irrational users from refusing to update their reputation key and exploiting the obsolete reputation keys to act maliciously. Besides, reputation key revocation would be an even greater challenge in the traditional TBE scheme where the trust rating mechanism relies on the collective opinions. This is the first reason why the reputation revocation approach should be provided in a TBE system.

In order to avoid the abuse of obsolete reputation key, both the encryption and decryption of the original TBE scheme uses the communication round t. Thus, a secure channel between a receiver and the TA is assumed to exist in order to guarantee the secure delivery of the fresh decryption key of the current communication round. Whenever a receiver needs to communicate with another nodes or access shared information encrypted with certain trust rating level, the receiver has to acquire the private key from the TA, which results in significant traffic burden to the TA (linearly dependent on the number of users n and the communication rounds between each communication pair). The workload at the receivers would also be heavy in this case because when the receivers receive the encrypted private decryption key from the TA, they have to decrypt the received message for the updated decryption before they could even start running the TBE decryption algorithm.

Our basic idea is to adopt the revocable IBE technique as the underlying tool to design our TBE scheme to mitigate the traffic between nodes and the TA for the private key delivery. In our TBE with reputation revocation (TBE-RR) scheme, we divide the time into fixed time periods just as in the underlying R-IBE system. The length of time period is design parameter and can be determined based on the statistics for the dynamic range of the user reputation gathered from the reputation systems[1]. At the beginning of the TBE-RR system activation, each user will first obtain a trust based private key from the TA. Compared with the original TBE scheme, a message will not only be encrypted under the recipient's identity and the range requirement, but also the current time period. Notice that the time period and communication round are different. In one time period, two communication pair might have gone through several rounds of communications and the receiver will use its decryption key for the current time period to decrypt the ciphertext. We differentiate the decryption key from the private key in this section. A user with a valid private key might not be able to generate a decryption key for a certain time period unless he/she is unrevoked in this very time period. The TA will periodically publish key update information only for the unrevoked user to generate the updated decryption key and will only need to deliver the updated private key for the revoked users when necessary (for instance, when the delivery is requested by the revoked users) so that the cost of distributing the updated private keys (in terms of communications and computations) would be reduced from $\mathcal{O}(n)$ to $\mathcal{O}(v)$, where v denotes the number of revoked users.

Our TBE-RR scheme can be considered as a combination of the R-IBE scheme and the original TBE scheme. At the initialization of our TBE-RR system, the TA runs **R-Setup** algorithm of the R-IBE scheme[1] constructs a binary tree with at least N leaf nodes, where $N = n \times \kappa$ and n is the number of system users. According to Sec.II, each user's trust rating value could be represented by a rating set consisting of at most κ identities. Therefore, the binary tree will cover all the rating sets S_{r_i} for each system user $id_i, i \in [1, n]$. The TA also holds the system public key pk and the master key

[1] Notice that all the algorithms containing a prefix **R-** in this section correspond to those algorithms in the R-IBE system introduced in Sec. IV-A

msk and reserves the memory space for messages, identity, time period and the empty revocation list rl after running **R-Setup** algorithm. Each user will be assigned with a private key which consists of all the identity based private keys with the corresponding trust rating set through running the **R-PriKeyGen**(msk, **id**) algorithm on each identity in the rating set. For example, for a system with four users id_1, id_2, id_3, id_4 with the respective rating value $\frac{1}{2}$, $\frac{1}{8}$, $\frac{1}{4}$, $\frac{1}{4}$, the TA will first construct a binary tree covering all the rating set as shown in Fig.3. To assign the private key for user id_2 with a rating value $\frac{1}{8}$, the TA will run **R-PriKeyGen**(mk, id) on each identity id belonging to the rating set $\{id_2||1, id_2||01, id_2||001\}$ and outputs $sk_{id_2||1}$, $sk_{id_2||01}$, $sk_{id_2||001}$, which constitute the user private key $sk_{id_2||r=\frac{1}{8}}$. In other words, the TA will assign each identity based private key according to the private key node set for all of these leaf nodes. We notice that the original TBE system is utilized in a different way in the sense that there is no temporal information contained in the user private key in our TBE-RR scheme. When the sender encrypts a message for receiver id_2 with a reputation range requirement $[0, \frac{1}{4}]$ in time period t, the message is encrypted under the range set $\{id_2||\top, id_2||0, id_2||01, id_2||010\}$ and time period t. Therefore, the sender executes the encryption algorithm of the revocable IBE scheme **R-Enc**(pk, id, t, M) on each identity id in the range set $\{id_2||\top, id_2||0, id_2||01, id_2||010\}$ to generate the ciphertext $\{C_{id_2||\top,t}(M), C_{id_2||0,t}(M), C_{id_2||01,t}(M), C_{id_2||010,t}(M)\}$, which constitute the final ciphertext $C_{id_2,[0,R],t}(M)$.

The TA publishes update information ku_t by running the update algorithm **R-KeyUpdate**(pk, msk, t, rl). This update information is only useful for those whose reputation value is not revoked. In other words, the receiver id_2 can successfully generate the decryption key for time period t only when its rating value is not revoked[2]. The receiver id_2 will execute **R-DecryKeyGen** algorithm, which takes as the input $sk_{id_2||r}$ and key update information ku_t to output the respective decryption key $dk_{id_2||r,t}$. This decryption key is composed of all the decryption keys for its rating set, i.e., $dk_{id_2||1,t}$, $dk_{id_2||01,t}$, $dk_{id_2||001,t}$. Thus, the receiver can run the **R-Dec** algorithm, which takes as the input the ciphertext $C_{id_2||01,t}(M)$ [3] and the respective decryption key $dk_{id_2||01,t}$ to recover the respective message M.

Complexity analysis: This construction results in a user private key of size $\mathcal{O}(\kappa \log N)$. However, the receiver, especially those users whose reputation stays stable over time, do not need to communicate with the TA. The only thing they need to do is to check out the published the update information periodically to generate their fresh decryption key for each new time period. This will significantly reduce the communication traffic between users and the TA. It will only add an additional group element to the ciphertext compared with the original TBE scheme if randomness reuse technique is adopted in the encryption algorithm [6]. If our TBE-RR scheme adopts the same underlying IBE technique as in the R-IBE system [2], then the sender in our TBE-RR system only needs to do an extra exponential computation when generating the ciphertext compared with that of the original TBE system. The workload of the TA will mainly be determined by the task of distributing the update information and delivering the private keys for the revoked users. The computation and communication costs of both tasks depend on the number of the revoked users.

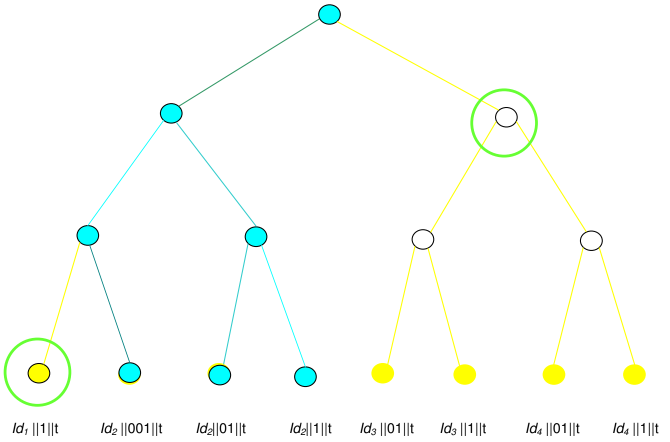

Fig. 3. Basic idea of our TBE-RR scheme: update information only cover those whose trust rating value has not changed

Security analysis: The security of our TBE-RR scheme can be reduced to the security of the underlying R-IBE scheme as stated in the following theorem. The proof is omitted due to the page limit.

Theorem 1: Suppose we have a sIND-CCA secure R-IBE scheme, then the TBE-RR scheme designed from this R-IBE scheme following our proposed procedure is also sIND-CCA secure.

V. Conclusion

In this paper, we have proposed to use revocable identity based encryption (R-IBE), to develop a novel trust based encryption (TBE) schemes used in information sharing and dissemination to significantly improve the efficiency in terms of communication overheads.

References

[1] Mudhakar Srivatsa, Shane Balfe, Kenneth G. Paterson, and Pankaj Rohatgi. Trust management for secure information flows. In *ACM Conference on Computer and Communications Security*, pages 175–188, 2008.

[2] Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. In *ACM Conference on Computer and Communications Security*, pages 417–426, 2008.

[3] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In *CRYPTO*, pages 213–229, 2001.

[4] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pages 47–53, 1984.

[5] Elaine Shi, John Bethencourt, Hubert T.-H. Chan, Dawn Xiaodong Song, and Adrian Perrig. Multi-dimensional range query over encrypted data. In *IEEE Symposium on Security and Privacy*, pages 350–364, 2007.

[6] Mihir Bellare, Alexandra Boldyreva, and Jessica Staddon. Randomness re-use in multi-recipient encryption schemeas. In *Public Key Cryptography*, pages 85–99, 2003.

[2] If the rating value r of this receiver is revoked at certain time period t', then the periodically published update information will only cover those leaf nodes except the rating set for $id_2||r$, i.e, all the yellow nodes.

[3] since $id_2||01$ is the intersection identity between the rating set and range set