# Exclusion-Intersection Encryption and Its Application to Searchable Encryption*

Sherman S.M. Chow**

Department of Computer Science
Courant Institute of Mathematical Sciences
New York University, NY 10012, USA
schow@cs.nyu.edu

**Abstract.** Identity (or identifier) based encryption has shown to be a useful cryptographic schema enabling secure yet flexible role-based access control. In this paper, we propose a new notion named as *exclusion-intersection encryption*: the sender can specify the targeted groups that are legitimated and interested in reading the documents in the encryption algorithm; there exists a trusted key generation centre generating the *intersection* private decryption keys on request. This special private key can only be used to decrypt the ciphertext which is of all the specified groups' interests, its holders are *excluded* from reading the documents targeted to any subset of the groups (e.g. the ciphertext of only a single group's interest). One of the applications of this new notion is to support an ad-hoc joint project of two groups which needs extra helpers that are not from either group.
Another interesting application of the proposed scheme is an encrypted audit log that supports conjunctive field keyword searching, which is the first in the literature.

**Key words:** access control, applied cryptography, audit log, searchable encryption

## 1 Introduction

Controlling the access of data via complex policies is always a challenging issue, especially for dynamic organizations where people assume different roles in different (possibly ad-hoc) projects and people's roles may change over time. Identity (or identifier) based encryption (e.g. [4, 6, 12, 16, 18, 25]) has shown to be a useful cryptographic schema enabling secure yet flexible role-based access control [1, 8, 19, 20, 23] (in particular, the access of the plaintext message encrypted in a ciphertext).

* A short note about a recent and independent work "Searchable Keyword-Based Encryption" by Park-Cha-Lee is given in the Appendix.

** Part of the research is done when the author is with Department of Computer Science, The University of Hong Kong.

In this paper, we propose a new notion named as *exclusion-intersection encryption*: the sender can specify the target groups (says $A$, $B$, and $C$) that are legitimated and interested to read the documents in the encryption algorithm; there exists a trusted key generation centre (KGC) generating *intersection* private decryption keys (e.g. $A \cap B \cap C$, $B \cap C$ or just $A$) on request. This special private key can only be used to decrypt the ciphertext which is of all the specified groups' interests (e.g. the decryption key of $A \cap B \cap C$ can decrypt the cipheretext which is of all of $A$ and $B$ and $C$'s interests), its holders are *excluded* from reading the documents targeted to any subset of this set of groups (e.g. the decryption key of $A \cap B \cap C$ can *neither* decrypt the ciphertext targeted to $A \cap B$, *nor* the ciphertext targetted to $C$). We use the "$\cap$" notation from the key's decryption power perspective: the key for $A \cap B$ is a less powerful key than the key for $A$, analogous to the fact that $A \cap B$ is a subset of $A$.

One of the applications of this new notion is to support an ad-hoc joint project of two groups which needs extra helpers that are not from either group. The KGC only only needs to generate the intersection private key to these extra helpers, then all parties concerned (both groups and those new helpers) can decrypt the documents for this joint project, but these new helpers cannot decrypt the documents which are confidential to each group. The key distribution is minimal as only these new helpers (instead of all related people of the project) need to get a new key. The scheme supports cryptographic workflow in the sense that sender can create the encrypted documents even the decryption key are yet to be generated by KGC and obtained by the related parties.

The proposed scheme is useful when the sender does not have the knowledge of the access-control policy nor the hierarchy of the groups in an organization. Consider an applicant for PhD programme who just gets a few more papers accepted for publication and wants to submit a more updated version of his curriculum vitae (CV) to a certain university so as to increase his chance of being admitted. The application committee usually consists of the staff members from both the graduate school and the department of interest, says Department of Computer Science (hereinafter referred as CS department). By using our proposed exclusion-intersection encryption, he can encrypt his CV to "*Graduate School*" $\cap$ "*CS Department*". As a result, the staff members at CS department, the staff members at graduate school, or a special group of people (hereinafter referred as "*Helpers*") only handling graduate admission of CS (if exists) can decrypt and read his CV, irrespective of the private key issuing policy of the university.

One may argue that it is possible to achieve the same things by using hierarchical encryption [4, 9, **?**,13, 16, 17, 25, 27]. However, notice that the sender may not know the hierarchy of the groups in that university (for examples, whether the graduate school is at a level higher than the CS department or if there is a group of people handling graduate admissions under the CS department), or actually there is no such hierarchy. One of the possible solution is that both of the graduate school and the CS department generate the "children private key" for CS department and graduate school respectively, i.e. the helpers will get both the

private key corresponding to "*Graduate School*" → "*CS Department*" and "*CS Department*" → "*Graduate School*" (where $A \rightarrow B$ denotes $A$ is at a level higher than $B$). It seems that the same result can be achieved as (1) the sender does not need to know the hierarchy (i.e. he can use either "*Graduate School*" → "*CS Department*" or "*CS Department*" → "*Graduate School*" as the identifier), (2) the helpers cannot read the existing encrypted document for "*Graduate School*" and "*CS Department*" (as being at the lower level of the hierarchy), (3) the KGC only needs to generate private key for the helpers. However, this solution is not scalable if the number of different groups involved increases.

Moreover, we can actually use this scheme in another way round. In normal hierarchical encryption, the one at a higher level of hierarchy (says the manager) has a higher decryption power (i.e. can decrypt the ciphertext designated to the users at a lower level of hierarchy, or his group of sub-ordinates). Now we consider the scenario that the privacy of sub-ordinates is of importance, such that their manager cannot read their private message unless the message is of whole group's interests. Suppose there is a group of students $\{ID_i | i = 1, 2, \cdots, t\}$ with a supervisor. One can assign the key $ID_1 \cap ID_2 \cdots \cap ID_t$ to the supervisor. In doing so, the supervisor cannot read a private message directed to oly one or a subgroup of students, but he can decrypt the encrypted messages when all students in his group are appointed as receivers. We remark that this is not a perfect application of our scheme. In contrast to the previous application, we need re-keying if some new members join the group.

Interestingly the proposed scheme can be applied on searching encrypted audit logs too. A secure audit log should be tamper-resistant and encrypted such that no adversary can modify nor read the audit log. On the other hand, it should be efficiently searchable by authorized auditors, yet the searching power delegated to the auditors should be limited. Our scheme gives rise to an encrypted audit log that supports conjunctive field keyword searching, which is the first in the literature.

## 2 Related Work

### 2.1 Access Control from Elliptic Curve Pairings

Notions similar to our proposed exclusion-intersection encryption can be found in [8] and [23], which considered the "conjunction" and "disjunction" of private keys associated with multiple identities. By conjunction, any one who has all the private keys involved with an encrypted message can do the decryption; while disjunction means any one who has at least one of the private keys involved with an encrypted message can get the plaintext. However, the papers provided neither security model nor formal proof. In a recent work [2], efficient multi-receiver identity-based encryption (i.e. encryption in "disjunction" model) was proposed together with a formal model and security proofs. However, to the best of authors' knowledge, there is no work addressing other special forms of access like the "exclusion-intersection" scenario considered in this paper.

### 2.2 Searchable Encryption of Audit Log

A closely related area of searchable encryption of audit log is searchable encryption in general. The majority of previous work in this area considers the scenario that a bandwidth constrained user, who stores documents on an untrusted server, delegates the searching power to that storage server. The question on how to perform searching on encrypted data was raised in [24], where a practical (in the sense of space and communication overhead) technique based on stream cipher was proposed as well. Their scheme works in a symmetric-key setting, which means the same key is used for encryption, decryption and searching of data.

Subsequently, index-based approaches [3, 7, 14] were proposed, where the restriction on the encryption mechanism of the document is removed. Basically, these work addressed the scenario where the encrypted data is stored in an untrusted remote server or any server outside the data owner's direct control (consider the scenarios of outsourcing the backup and the storage of enterprise's data to data warehousing companies and storage service for client to retrieve data using his/her wireless PDA) , and the data owner wants to retrieve only the data satisfied with certain criteria (instead of the whole set of data) from the server when necessary, without delegating the decryption power or any power of distinguishing other encryption documents (except the above mentioned criteria). All these schemes are symmetric in the sense that the data owner builds the index.

The first public key scheme for keyword search over encrypted data were presented in [5], where every body can use the public key to create the encrypted indexes such that only the one holding the corresponding private key can generate trapdoor to search over these index. The schemes are made possible with the use of bilinear pairings. Subsequently, a pairing-based scheme for secure conjunctive keyword search over data encrypted by symmetric key was proposed in [15]. Their work pointed out the issue that the trivial solution of conjunctive keyword search using set intersection of two simple keyword search queries is insecure (as extra information about which set of encrypted documents match only one of the criteria will be leaked) and the use of meta-keyword defining for every possible conjunction of keywords are obviously unsatisfactory. Public key encryption with conjunctive field keyword search was recently proposed in [22].

A closely related work [26] proposed a searchable encrypted audit log system. Each record is encrypted with a random key, this key is which is in turn encrypted using identity-based encryption with each keyword as the identity (public key). A time-scoped searchable encrypted audit log system was subsequently proposed in [10], which can be regarded as building on top of the idea in [26]. Time-scoped searching is accomplished by storing a backpointer for each keyword which indicates the most recently logged record containing that keyword. At the same time the log server periodically creates a special kind of logs, namely anchor logs, which serve as boundaries for demarcating the time scopes of the records. Although [10] equipped a searchable encrypted audit log system with time-scoped searching, their construction cannot be trivially extended to support

conjunctive keyword search as extensive use of identity-based encryption may result. On the other hand, our solution can be easily extended to support time-scoped searching by making the logging time as one of the searchable fields.

One may be tempted to use the conjunction version in [23] to build an audit log with conjunctive field keyword search. However, no formal treatment has been made to investigate the possibility and the security of this approach. Even it is possible, it provides a less efficient solution when compared with our proposed scheme.

## 3 Building Blocks

### 3.1 Bilinear Pairings

Bilinear pairing is an important primitive for many cryptographic schemes. In particular, many access control schemes and searchable encryption are based on elliptic-curve pairings [1, 2, 5, 8, 10, 15, 22, 23, 26]. Here, we describe some of its key properties.

Let $(\mathbb{G}_1, +)$ and $(\mathbb{G}_2, \cdot)$ be two cyclic groups of prime order $q$. The bilinear pairing is given as $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, which satisfies the following properties:

1. *Bilinearity*: For all $P, Q, R \in \mathbb{G}_1$, $\hat{e}(P+Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$, and $\hat{e}(P, Q+R) = \hat{e}(P, Q)\hat{e}(P, R)$.
2. *Non-degeneracy*: There exists $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P, Q) \neq 1$.
3. *Computability*: There exists an efficient algorithm to compute $\hat{e}(P, Q)\ \forall P, Q \in \mathbb{G}_1$.

We assume the existence of a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the property that the following problem is hard to compute.

**Definition 1.** *Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order $q$, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator $P$ of $\mathbb{G}_1$, the q-Decisional Bilinear Diffie-Hellman Inversion Problem (q-DBDHIP) in $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ is to decide whether $R = \hat{e}(P, P)^{1/x}$ given $(P, xP, x^2P, \cdots, x^qP)$ and an element $R \in \mathbb{G}_2$.*

### 3.2 Symmetric Encryption

Let $\{\mathcal{E}_{(\cdot)}(\cdot), \mathcal{D}_{(\cdot)}(\cdot)\}$ be a pair of symmetric encryption and decryption function with key space $\mathbb{K}$. We requires that the symmetric encryption functions are find-guess secure [11], i.e. any polynomially bounded adversary cannot distinguish between $\mathcal{E}_K(m_0)$ and $\mathcal{E}_K(m_1)$ for any $K \in \mathbb{K}$ with a probability significantly greater than $1/2$, where $m_0$ and $m_1$ are two messages of the equal length chosen by the adversary. Note that the adversary has neither encryption oracle access nor decryption oracle access.

### 3.3 Hash Functions

Our schemes will employ the following cryptographic hash functions, which we assume are modelled by random oracles in our security proof:

- $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$
- $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$
- $H_3 : \mathbb{G}_2 \rightarrow \mathbb{K}$

  where $\mathbb{K}$ is the key space of the symmetric encryption function.

## 4 Exclusion-Intersection Encryption

### 4.1 Framework

- $\texttt{Setup}(1^k)$: On an unary string input $1^k$ where $k$ is a security parameter, it produces the master secret key $\mathcal{S}$ and the common public parameters *params*, which include a description of a finite plaintext space and a description of a finite ciphertext space. We omitted the inclusion of common public parameters as part of the input in the descriptions of the remaining algorithms.
- $\texttt{Trapdoor}(\mathcal{S}, \{Q_i\})$: Taking $\{Q_i\}$ as the input of a single group or a list of groups, it produces a trapdoor $T_{\{Q_i\}}$ with the help of the master secret key $\mathcal{S}$, the resulting trapdoor is the private key for a single group or an "intersection private key" for the intersection of all the specified groups, depending on the number of groups specified in $\{Q_i\}$).
- $\texttt{EIE}(m, \{W_i\})$: For a plaintext message $m$ together with a list of targeted groups $\{W_i\}$, it produces an exclusion-intersection encryption $S_{\{W_i\}}$ of $m$.
- $\texttt{Decrypt}(S, T_{\{Q_i\}})$: Given the exclusion-intersection encryption $S_{\{W_i\}}$ of $m$, if the group associated with the trapdoor $T_{\{Q_i\}}$ is a subset of the targeted groups associated with $S_{\{W_i\}}$, i.e. $\{Q_i\} \subseteq \{W_i\}$, then outputs $m$, '$\perp$' otherwise.

### 4.2 Security

Here we consider the de-facto standard of a secure public key encryption scheme, which is indistinguishability against adaptive chosen-ciphertext-and-identifier attacks. For our exclusion-intersection encryption, security is defined by the following IND-EIE-CCIA2 game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

*Setup*: The challenger $\mathcal{C}$ takes a security parameter $k$ and runs `Setup` to generate common public parameters *param* and the master secret key $\mathcal{S}$. $\mathcal{C}$ sends *param* to $\mathcal{A}$.

*Phase 1*: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries). The types of queries allowed are described below.

- `Trapdoor`: $\mathcal{A}$ chooses a list of groups $\{Q_i\}$, $\mathcal{C}$ computes $\texttt{Trapdoor}(\mathcal{S}, \{Q_i\})$ and sends the result to $\mathcal{A}$.
- `Decrypt`: $\mathcal{A}$ chooses a ciphertext $S$, $\mathcal{C}$ computes a certain trapdoor that can decrypt $S$ according to the auxiliary information $\mathcal{L}$ contained in $S$, decrypt the ciphertext $S$ and sends the resulting plaintext $m$ or the symbol $\perp$ to $\mathcal{A}$

*Challenge*: The adversary $\mathcal{A}$ decides when Phase 1 ends. Then, it outputs two equal length plaintexts, $m_0$ and $m_1$, and a set of group identifiers $\{ID_i\}_{i=1,2,\cdots,t}$ on which it wishes to be challenged. The set $\{ID_i\}_{i=1,2,\cdots,t}$ *or a subset of it* should not appear in any `Trapdoor` queries in Phase 1. The challenger $\mathcal{C}$ picks a random bit $b$ from $\{0,1\}$, computes $S = \texttt{EIE}\ (m_b, \{ID_i\}_{i=1,2,\cdots,t})$ and returns $S$ to $\mathcal{A}$.

*Phase 2*: The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the similar restriction on `Trapdoor` query and the restriction that a `Decrypt` query to obtain the plaintext for $S$ cannot be made.

*Guess*: The adversary $\mathcal{A}$ has to output a guess $b'$. It wins the game if $b' = b$.

The *advantage* of $\mathcal{A}$ is defined as $Adv(\mathcal{A}) = |2P[b' = b] - 1|$ where $P[b' = b]$ denotes the probability that $b' = b$.

**Definition 2.** *An exclusion-intersection encryption scheme is said to have the indistinguishability against adaptive chosen-ciphertext-and-identifier attacks property (IND-EIE-CCIA2 secure) if no adversary has a non-negligible advantage in the IND-EIE-CCIA2 game.*

Notice that the definition of IND-EIE-CCIA2 game is kind of similar to that of ILCR (indistinguishability of limited ciphertext from random) game defined in [22], which is essential for the security of the encrypted searchable audit log. Instead of choosing an identifier to be challenged, the adversary chooses the keyword in a certain position in the ILCR game. This challenge keyword is associated with either one of the messages $m_0$ and $m_1$, and the goal of the adversary is to distinguish these two messages without asking for the trapdoor that can distinguish $m_0$ from $m_1$, which matches the restriction of the adversary in IND-EIE-CCIA2 game that no trapdoor queries on the challenge identifier is allowed. Moreover, we provide decryption queries, which are not applicable in their scenario.

## 5 Proposed Construction

The key generation centers executes the `Setup` algorithm at the first place and generates the trapdoor $T_{\{}Q_i\}$ for the group of users $\{Q_i\}$ using the `Trapdoor` algorithm. Anyone can use the `EIE` algorithm to encrypt a message $m$ for the appointed recipients $\{W_i\}$. Finally, one holding the trapdoor $T_{\{}Q_i\}$ can decrypt the ciphertext if $\{Q_i\} \subseteq \{W_i\}$.

- $\texttt{Setup}(1^k)$: Let $q$ be the order of the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ which is determined by the security parameter $k$. The algorithm chooses random numbers $s_1$, $s_2$ and $s_3 \in \mathbb{Z}_p$ and a generator $P$ of $\mathbb{G}_1$. It outputs $params = [P, Y = s_1P, Z_1 = s_2P, Z_2 = s_3P, g = \hat{e}(P,P)]$ and $\mathcal{S} = [s_1, s_2, s_3]$.
- $\texttt{Trapdoor}(\mathcal{S}, \{Q_i\}_{i=1,2,\cdots,t})$: Selects a random number $u \in \mathbb{Z}_p$ and makes $T_{\{Q_i\}_{i=1,2,\cdots,t}} = [T_1, T_2, T_3]$ where
  - $T_1 = (\frac{1}{t \cdot s_1 + H_1(Q_1) + \cdots + H_1(Q_t) + s_3 u})P$
  - $T_2 = \frac{1}{s_2} T_1$
  - $T_3 = u$.
- $\texttt{EIE}(m, \{W_i\}_{i=1,2,\cdots,n})$:
  1. Selects random numbers $r_1, \cdots, r_n \in \mathbb{Z}_p$.
  2. Computes $\{B_1, \cdots, B_n\} = \{r_1 Z_1, \cdots, r_n Z_1\}$.
  3. Computes $r_0 = H_2(m||B_1|| \cdots ||B_n)$, where each $B_i$ is treated as a bit string and $||$ represents the string concatenation operator.
  4. Computes $C = r_0 Z_2$.
  5. Computes $\mathcal{K} = H_3(g^{r_0})$.
  6. Encrypts message by $E = \mathcal{E}_{\mathcal{K}}(m)$.
  7. Computes $\{A_1, \cdots, A_n\} = \{r_0(Y + H_1(W_1)P) + r_1 P, \cdots, r_0(Y + H_1(W_n)P) + r_n P\}$.
  8. Outputs $\texttt{EIE}(m, \{W_i\}) = [A_1, \cdots, A_n, B_1, \cdots, B_n, C, E, \mathcal{L}]$ where $\mathcal{L}$ is a label that contains information about how "$A_i$" is associated with each group.
- $\texttt{Decrypt}(S, T_{\{Q_i\}})$: Let $S = [A_1, \cdots, A_n, B_1, \cdots, B_n, C, E, \mathcal{L}]$, suppose $I_1, I_2, \cdots, I_t$ are the positions of the groups from $\{Q_i\}$ in the list $\{W_i\}$ specified by the auxiliary information $\mathcal{L}$.
  1. Computes $\mathcal{K}' = \frac{\hat{e}(A_{I_1} + \cdots + A_{I_t} + T_3 C, T_1)}{\hat{e}(B_{I_1} + \cdots + B_{I_t}, T_2)}$.
  2. Recover $m' = \mathcal{D}_{(H_3(K'))}(E)$.
  3. If $\mathcal{K}' = H_3(g^{H_2(m'||B_1|| \cdots ||B_n)})$, output $m'$; otherwise output $\perp$.

### 5.1 Analysis

We first give the motivation of the choice of hash function for hashing the group identifier. Similar to [22], the advantage of such choice is that computationally expensive admissible encoding scheme hashing to $\mathbb{G}_1$ is not needed [6]. The construction is efficient in the sense that no pairing operation is needed for the generation of trapdoor and encryption while it only takes two pairing operations for decrypt operation.

For the efficiency of searchable encrypted audit log, both encryption and decryption require one pairing operation in the solution of [26], while our solution removed the computation of pairing from the encryption. Notice that pairing operation is still needed if one use the conjunction version in [23] to build an audit log with conjunctive field keyword search, as described in our review section.

The following theorem summarizes the security of our proposed scheme.

**Theorem 1** *In the random oracle model (the hash functions are modeled as random oracles), we assume that we have an adversary $\mathcal{A}$ that is able to win the IND-EIE-CCIA2 game (i.e. $\mathcal{A}$ is able to distinguish ciphertexts given by the challenger), with an advantage $\epsilon$ when running in a time $t$ and asking at most $q_H$ identifier hashing queries, at most $q_T$ trapdoor generation queries, at most $q_R$ $H_3$ queries, and $q_D$ decryption queries. Then, there exists a distinguisher $\mathcal{C}$ that can solve the $(q_T+1)$-DBDHIP with non-negligible probability.*

*Proof.* On input of $(P, xP, x^2P, \cdots, x^{q_T+1}P, R)$, $\mathcal{C}$'s goal is to check whether $R = \hat{e}(P,P)^{1/x}$. We firstly describe the simulation of $\mathcal{C}$.
`Setup`:

1. Chooses $\zeta_1, \zeta_2, \cdots, \zeta_{q_T} \in_R \mathbb{Z}_p^*$.
2. Expands the term in $f(z) = \prod_{j=1}^{q_T}(z+\zeta_j)$ by $f(z) = \sum_{i=0}^{q_T+1} c_i z^i$.
3. Computes $U = f(x)P$ by $\sum_{i=0}^{q_T+1} c_i x^i P$ and $V = xU$ by $\sum_{i=0}^{q_T+1} c_{i-1} x^i P$.
4. For $1 \le i \le q_T$, computes $\frac{1}{x+\zeta_i}U = \frac{f(x)}{x+\zeta_i}P = \sum_{j=0}^{q_T-1} d_j x^j P$, these values are stored to be used in handling the `Trapdoor` queries.
5. Computes $R_U = R^{{c_0}^2} \cdot \hat{e}((c_0 + \sum_{i=0}^{q_T} c_i x^i)P, \sum_{i=0}^{q_T-1} c_{i+1} x^i P)$. If $R$ is indeed the solution to the $(q_T+1)$-computational bilinear Diffie-Hellman inversion problem (i.e. $R = \hat{e}(P,P)^{1/x}$), we have $R_U = \hat{e}(U,U)^{1/x}$, this value will be embedded in the challenge ciphertext.
6. Chooses $\alpha, \beta, \gamma_1, \gamma_2 \in_R \mathbb{Z}_p^*$.
7. Computes $Y = \alpha - \beta U$.
8. Compute $Z_1 = \gamma_1 U$.
9. Compute $Z_2 = \gamma_2 V$.
10. Outputs $params = [U, Y, Z_1, Z_2, g = \hat{e}(U,U)]$. Notice that all part of the corresponding master secret key $\mathcal{S}$ ( $[s_1 = \alpha x - \beta, s_2 = \gamma_1, s_3 = \gamma_2 x]$) are unknown $\mathcal{C}$ except $s_2$.

$H_1$ queries: Roughly speaking, these answers are randomly generated, but to maintain the consistency and to avoid collision, $\mathcal{C}$ keeps the list $L_1$ to store the answers used (i.e. the same answer will be returned if the query has been made before). Besides, some extra information about the answers returned will be stored in the list too. Suppose $H_1(W_i)$ has not been asked,

1. Randomly chooses $c_i \in_R \{0,1\}$, where the probability for $c_i = 0$ is $\tau$ (to be determined).
2. If $c_i = 0$, randomly chooses $h_i \in_R \mathbb{Z}_p^*$, otherwise set $h_1 = \beta$. For either way, $h_i$ will be returned as the answer and $< W_i, h_i, c_i >$ is stored in list $L_1$.

$H_2$ and $H_3$ queries: When $\mathcal{A}$ asks queries on these hash values, $\mathcal{C}$ checks the respective list $L_2$ or $L_3$. If an entry for the query is found, the same answer will be given to $\mathcal{A}$; otherwise, a randomly generated value will be used as an answer to $\mathcal{A}$, the query and the answer will then be stored in the list.

`Trapdoor` queries: Suppose $\mathcal{A}$ asks for the trapdoor corresponding to the group $\{Q_i\}_{i=1,2,\cdots,t} = \{Q_{i,1}, Q_{i,2}, \cdots, Q_{i,t}\}$.

1. Get the entries $< Q_{i,j}, h_{i,j}, c_{i,j} >$ in $L_1$, if $c_{i,j} = 1\ \forall j$, aborts the simulation.
2. Computes $E_i = ts_1 + h_{i,1} + h_{i,2} + \cdots + h_{i,t} = t\alpha x + (h_{i,1} + h_{i,2} + \cdots + h_{i,t} - t\beta)$. For simplicity we let $\alpha^* = t\alpha$ and $\beta^* = (h_{i,1} + h_{i,2} + \cdots + h_{i,t} - t\beta)$, so $E_i$ can be expressed as $\alpha^* x + \beta^*$.
3. Picks $i$-th pair $(\zeta_i, \frac{1}{x+\zeta_i} U)$ generated at the `Setup` phase.
4. Computes $u_i = ((\beta^*/\zeta_i - \alpha^*)/\gamma_2)$ and $v_i = \beta^*/\zeta_i$ (such that the equation $\frac{1}{(x+\zeta_i)} = \frac{v_i}{\alpha^* x + \beta^* + \gamma_2 x u_i}$ holds).
5. Compute $F_i = \frac{1}{v_i(x+\zeta_i)}$ (which is equal to $\frac{1}{\alpha^* x + \beta^* + \gamma_2 x u_i}$).
6. Output the trapdoor as $[F_i, \frac{1}{\gamma_1} F_i, u_i]$ (which is valid since $\frac{1}{\alpha^* x + \beta^* + \gamma_2 x u_i} = \frac{1}{E_i + s_3 u_i}$).

`Decrypt` queries: Suppose $\mathcal{A}$ asks for the decryption of the ciphertext $S_i$ corresponding to the group $\{Q_i\}_{i=1,2,\cdots,t} = \{Q_{i,1}, Q_{i,2}, \cdots, Q_{i,t}\}$.

1. Get the entries $< Q_{i,j}, h_{i,j}, c_{i,j} >$ in $L_1$, if there exists $j$ such that $c_{i,j} = 0$, generates the trapdoor using the above simulation and decrypt the ciphertext.
2. Suppose $S_i = [A_{i,1}, A_{i,2}, \cdots, A_{i,n}, B_{i,1}, B_{i,2}, \cdots, B_{i,n}, C_i, E_i, \mathcal{L}_i]$, and $I_1, I_2, \cdots, I_t$ are the positions of the groups from $\{Q_i\}$ in the list $\{W_i\}$ specified by the $\mathcal{L}$.
3. Picks $A_{I_1}$, $B_{I_1}$ and $C_i$ which are in the form of $r_0 Y + r_0 H_1(W_{I_1}) U + r_1 U$, $r_1 Z_1 = r_1 \gamma_1 U$ and $r_0 Z_2 = r_0 \gamma_2 V$ respectively. Compute $r_0 U = \frac{1}{(H_1(W_{I_1}) - \beta)} [A_{I_1} - \frac{1}{\gamma_1} B_{I_1} - \frac{\alpha}{\gamma_2} C_i]$.
   The consistency can be shown by

$$\begin{aligned}
&(H_1(W_{I_1}) - \beta)^{-1} [A_{I_1} - \frac{1}{\gamma_1} B_{I_1} - \frac{\alpha}{\gamma_2} C_i] \\
&= (H_1(W_{I_1}) - \beta)^{-1} [r_0 Y + r_0 H_1(W_{I_1}) U - r_0 \alpha V] \\
&= (H_1(W_{I_1}) - \beta)^{-1} [r_0(\alpha V - \beta U) + r_0 H_1(W_{I_1}) U - r_0 \alpha V] \\
&= (H_1(W_{I_1}) - \beta)^{-1} [-r_0 \beta U + r_0 H_1(W_{I_1}) U] \\
&= (H_1(W_{I_1}) - \beta)^{-1} [r_0 (H_1(W_{I_1}) - \beta) U] \\
&= r_0 U
\end{aligned}$$

4. Recover $m' = \mathcal{D}_{(H_3(\hat{e}(r_0 U, U)))}(E_i)$.
5. If $H_2(m' || B_1 || \cdots || B_n) U \neq r_0 U$, return $\perp$. Otherwise return $m'$.

`Challenge`: Eventually adversary $\mathcal{A}$ produces a list of group identifier $\{ID_i | i = 1, 2, \cdots, t\}$ and a pair of message $m_0, m_1$ on which it wishes to be challenged. The challenge is generated as follows.

1. Pick a random bit $b$.
2. Get $< ID, h_i, c_i >$ from list $L_1$. If $c_i \neq 1$, $\mathcal{C}$ aborts.
3. Select random $\rho$, $r_1, \cdots, r_m \in \mathbb{Z}_p$.
4. Compute $A_i = \rho U + r_i U$ and $B_i = r_i Z_1$ for $1 \leq i \leq t$, $C = \rho s_3 U$, $E = \mathcal{E}_{(H_3(R_U{}^{\rho}))}(m_b)$, and $\mathcal{L} = \{ID_i\} i = 1, 2, \cdots, t$. It is easy to see that the ciphertext is valid.

Notice that $\mathcal{C}$ does not know the value of "$r_0$" value associated with this challenge, which may make the simulation of $H_2$ unfaithful. However, it is of negligible probability that $\mathcal{A}$ will make such query. Furthermore, by memorizing the value of $\rho$, $\mathcal{C}$ can always check whether the output of $H_2$ is the $x$ associated with the underlying problem.

`Output`: Finally $\mathcal{A}$ outputs a bit $b'$. If $b' = b$, return "true", "false" otherwise. □

# 6 Audit Logs Encryption with Conjunctive Field Keyword Search

## 6.1 Relationship with Exclusion-Intersection Encryption

Our proposed exclusion-intersection encryption can be utilized to support audit logs encryption with conjunctive field keyword search. The idea is to use each identifier to act as a "`field_name` = `value`" pair. A subtle difference is that the groups involved are different for each time for exclusion-intersection encryption, so it is necessary to include the auxiliary information $\mathcal{L}$ which enables one to take out the appropriate piece of ciphertext for decryption. In an audit log system, the keywords to be searched for are fixed for each system. Instead of including unnecessary auxiliary information $\mathcal{L}$ in the ciphertext, the position of the searchable field supported by the trapdoor are included when the trapdoor is delegated to the auditor. Indeed, the auxiliary information $\mathcal{L}$ must not be included in the ciphertext for the encrypted audit log scenario, since every one can know the content of the log entry by simply looking up the name and value of the field from this piece of information.

## 6.2 System Design

Our scheme shares a similar design as the construction in [26] to devise an encrypted audit logs that support conjunctive field keyword searching.

**Setup:** In our system, the private key is $\mathcal{S}$ and the public key is *params* which is generated by `Setup`. The private key and the public key are distributed to the audit escrow agent and the logging server respectively.

**Encryption:** When the logging server wants to encrypt a log entry $m$ with a list of keywords $\{W_i\}$, it runs `EIE` to generate the encrypted searchable indexes $S$, which is in turn stored in the logging server as an entry in the audit log.

**Search and Decryption:** When an investigator wants to search the entry which contains certain keywords $\{Q_i\}$, he sends the list of keywords to the audit escrow agent. If the audit escrow agent approves the requests, the agent in turn produces the trapdoor $T_{\{Q_i\}}$ using the private key $\mathcal{S}$, the target keywords and their positions in log entries. Then for each entry, `Decrypt` will be used to get all the related log entries.

## 7 Conclusion

We introduce the notion of *Exclusion-Intersection Encryption*, with a concrete construction, which provides a flexible solution for the access control of the plaintext message encrypted in a ciphertext. We illustrate two sample scenarios that exclusion-intersection encryption provides a better solution than traditional PKI-based schemes and hierarchical identity-based encryption schemes. As a bonus result, our scheme gives rise to an encrypted audit log that supports conjunctive field keyword searching, which is the first in the literature. Our scheme is provably secure under the random oracle model, assuming the hardness of the decisional bilinear Diffie-Hellman problem. We believe that exclusion-intersection encryption will give rise to other innovative applications other than those described in the paper.

## Acknowledgement

The author would like to thank the anonymous reviewers of CT-RSA 2006 for pointing out the mistake of the preliminary version of this paper. The author is also grateful to Dr. S.M. Yiu for giving helpful suggestions, Ms. Pierre K.Y. Lai for her help in the preparation of this paper, and Mr. Boris W.S. Yiu for naming the proposed notion.

## References

1. S.S. Al-Riyami, J. Malone-Lee, and N.P. Smart. Escrow-Free Encryption Supporting Cryptographic Workflow. Cryptology ePrint Archive, Report 2004/258, 2004. Available at http://eprint.iacr.org.
2. Joonsang Baek, Reihaneh Safavi-Naini, and Willy Susilo. Efficient Multi-receiver Identity-Based Encryption and Its Application to Broadcast Encryption. In Serge Vaudenay, editor, *Public Key Cryptography - PKC 2005, 8th International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23-26, 2005, Proceedings*, volume 3386 of *Lecture Notes in Computer Science*, pages 380–397. Springer, 2005.
3. Steven M. Bellovin and William R. Cheswick. Privacy-Enhanced Searches using Encrypted Bloom Filters. Cryptology ePrint Archive, Report 2004/022, 2004. Available at http://eprint.iacr.org.
4. Dan Boneh and Xavier Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 2004.
5. Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public Key Encryption with Keyword Search. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, volume 3027 of *Lecture Notes in Computer Science*, pages 506–522. Springer, 2004.

6. Dan Boneh and Matt Franklin. Identity-Based Encryption from the Weil Pairing. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag Heidelberg, 2001.
7. Yan-Cheng Chang and Michael Mitzenmacher. Privacy Preserving Keyword Searches on Remote Encrypted Data. In John Ioannidis, Angelos D. Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security, Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005, Proceedings*, volume 3531 of *Lecture Notes in Computer Science*, pages 442–455, 2005. Also available at Cryptology ePrint Archive, Report 2004/051.
8. L. Chen, Keith Harrison, David Soldera, and Nigel P. Smart. Applications of Multiple Trust Authorities in Pairing Based Cryptosystems. In George I. Davida, Yair Frankel, and Owen Rees, editors, *Infrastructure Security, International Conference, InfraSec 2002 Bristol, UK, October 1-3, 2002, Proceedings*, volume 2437 of *Lecture Notes in Computer Science*, pages 260–275. Springer, 2002.
9. Eu-Jin Goh Dan Boneh, Xavier Boyen. Hierarchical Identity Based Encryption with Constant Size Ciphertext. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456. Springer, 2005.
10. Darren Davis, Fabian Monrose, and Michael K. Reiter. Time-Scoped Searching of Encrypted Audit Logs. In Javier Lopez, Sihan Qing, and Eiji Okamoto, editors, *Information and Communications Security, 6th International Conference, ICICS 2004, Malaga, Spain, October 27-29, 2004, Proceedings*, volume 3269 of *Lecture Notes in Computer Science*, pages 532–545. Springer-Verlag, 2004.
11. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure Integration of Asymmetric and Symmetric Encryption Schemes. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 537–554. Springer, 1999.
12. David Galindo. Boneh-Franklin Identity Based Encryption Revisited. In *Automata, Languages and Programming: 32nd International Colloquium, ICALP 2005, Lisboa, Portugal, July 11-15, 2005. Proceedings*, volume 3580 of *Lecture Notes in Computer Science*. Springer, 2005.
13. Craig Gentry and Alice Silverberg. Hierarchical ID-Based Cryptography. In Yuliang Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566. Springer, 2002. Available at http://eprint.iacr.org.
14. Eu-Jin Goh. Secure Indexes. Cryptology ePrint Archive, Report 2004/016, 2004. Available at http://eprint.iacr.org.
15. Philippe Golle, Jessica Staddon, and Brent R. Waters. Secure Conjunctive Keyword Search over Encrypted Data. In Markus Jakobsson, Moti Yung, and Jianying Zhou, editors, *Applied Cryptography and Network Security, Second International Conference, ACNS 2004, Yellow Mountain, China, June 8-11, 2004, Proceedings*, volume 3089 of *Lecture Notes in Computer Science*, pages 31–45. Springer, 2004.

16. Yumiko Hanaoka, Goichiro Hanaoka, Junji Shikata, and Hideki Imai. Identity-Based Hierarchical Strongly Key-Insulated Encryption and Its Application. In *Advances in Cryptology - ASIACRYPT 2005, 11th International Conference on the Theory and Application of Cryptology and Information Security, Chennai (Madras), India, December 4-8, 2005, Proceedings*, 2005. Also available at Cryptology ePrint Archive, Report 2004/338.
17. Jeremy Horwitz and Ben Lynn. Toward Hierarchical Identity-Based Encryption. In Lars R. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings*, volume 2332 of *Lecture Notes in Computer Science*, pages 466–481. Springer, 2002.
18. Benoit Libert and Jean-Jacques Quisquater. Identity Based Encryption Without Redundancy. In John Ioannidis, Angelos D. Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security, Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005, Proceedings*, volume 3531 of *Lecture Notes in Computer Science*, pages 285–300, 2005.
19. Marco Casassa Mont and Pete Bramhall. IBE Applied to Privacy and Identity Management. Technical Report HPL-2003-101, Trusted Systems Laboratory, Hewlett-Packard Laboratories, 2003.
20. Marco Casassa Mont, Pete Bramhall, and Chris R. Dalton. A Flexible Role-based Secure Messaging Service: Exploiting IBE Technology in a Health Care Trial. Technical Report HPL-2003-21, Trusted Systems Laboratory, Hewlett-Packard Laboratories, 2003.
21. Dong Jin Park, Juyoung Cha, and Pil Joong Lee. Searchable Keyword-Based Encryption. Cryptology ePrint Archive, Report 2005/367, 2005. Available at http://eprint.iacr.org.
22. Dong Jin Park, Kihyun Kim, and Pil Joong Lee. Public Key Encryption with Conjunctive Field Keyword Search. In Chae Hoon Lim and Moti Yung, editors, *Information Security Applications: 5th International Workshop, WISA 2004, Jeju Island, Korea, August 23-25, Revised Selected Papers*, volume 3325 of *Lecture Notes in Computer Science*, pages 73–86. Springer-Verlag, 2004.
23. Nigel P. Smart. Access Control Using Pairing Based Cryptography. In Marc Joye, editor, *Topics in Cryptology - CT-RSA 2003, The Cryptographers' Track at the RSA Conference 2003, San Francisco, CA, USA, April 13-17, 2003, Proceedings*, volume 2612 of *Lecture Notes in Computer Science*, pages 111–121. Springer, 2003.
24. Dawn Xiaodong Song, David Wagner, and Adrian Perrig. Practical Techniques for Searches on Encrypted Data. In *Proceedings of the 2000 IEEE Symposium on Security and Privacy (S&P 2000)*, pages 44–55. IEEE, 2000.
25. Brent Waters. Efficient Identity-Based Encryption Without Random Oracles. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer, 2005.
26. Brent R. Waters, Dirk Balfanz, Glenn Durfee, and Diana K. Smetters. Building an Encrypted and Searchable Audit Log. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2004, San Diego, California, USA*. The Internet Society, 2004.
27. Danfeng Yao, Nelly Fazio, Yevgeniy Dodis, and Anna Lysyanskaya. ID-based Encryption for Complex Hierarchies with Applications to Forward Security and Broadcast Encryption. In *CCS '04: Proceedings of the 11th ACM conference on*

*Computer and Communications Security*, pages 354–363, New York, NY, USA, 2004. ACM Press.

## A Attacking Park-Cha-Lee's Searchable Keyword-Based Encryption

In an independent and possibly subsequent work [21], the notion of *Searchable Keyword-Based Encryption* was proposed. This work is more or less the same as the application of our exclusion-intersection encryption in supporting searchable encrypted audit-log, and shares a similar construction with ours. However, their scheme does not satisfy their definition of security, in particular, their scheme is distinguishable under the chosen ciphertext attack.

The attack is outlined as follows. After obtained the challenge ciphertext, which is in the form $[U, A_1, A_2, \cdots, A_m, B_1, B_2, \cdots, B_m, C, S, R]$; the adversary prepares a ciphertext $[U, A_1, A_2, \cdots, A'_m, B_1, B_2, \cdots, B'_m, C, S, R]$ to the decryption oracle where

- $A'_m = rP_1$, and
- $B'_m = rY_{m+1} = rs_{m+1}P_1$ where $r \in \mathbb{Z}_p^*$ is randomly chosen.

The decryption oracle first computes $\tilde{h^{r_0}} = \frac{\hat{e}(A_1+\cdots+A'_m+{T_3}^D C, {T_1}^D)}{\hat{e}(B_1+\cdots+B'm, {T_2}^D)}$. Notice that ${T_1}^D = s_{m+1}{T_2}^D$ always hold in their scheme. The addition of component cancels each other since

$$\begin{aligned}\hat{e}(rP_1, {T_1}^D) &= \hat{e}(rP_1, {T_1}^D) \\ &= \hat{e}(rP_1, s_{m+1}{T_2}^D) \\ &= \hat{e}(rs_{m+1}P_1, {T_2}^D) \\ &= \hat{e}(rY_{m+1}, {T_2}^D).\end{aligned}$$

As a result, $\tilde{h^{r_0}}$ is the value to be calculated by the decryption oracle for the *challenge ciphertext*, the rest of the decryption algorithm will use this ephemeral value and do the decryption for the adversary. Since the rest of the step does not depends on the modified value of $A'_m$ and $B'_m$, the decryption result is simply the decryption of the *challenge ciphertext*. Notice that it corresponds to the adaptive "test" queries if only searching but not decryption is considered. The source of this flaw may due to the lack of adaptive "test" queries in previous security model of public key encryption with keyword search like the one defined in [22].