A Symantic Public Key Encryption with Midden Field for Keyword Search

Manigandan. S. K\textsuperscript{1} \& Sethupathi. M\textsuperscript{2}

\textsuperscript{1}Assistant Professor, \textsuperscript{2}PG Scholar

Department of MCA. Vel Tech High Tech Dr.Rangarajan Dr.Sakunthala Engineering College, Avadi, Chennai-62.

1. Abstract: A lot of interest has been recently into public-key encryption with keyword search (PEKS), which keeps public-key encrypted documents amendable to secure keyword search. However, PEKS resist against keyword guessing attack by assuming that the size of the keyword space is beyond the polynomial level. We present the first efficient Identity-Based Encryption (IBE) scheme that is fully secure without random oracles. We first present our IBE construction and reduce the security of our scheme to the decisional Bilinear Diffie-Hellman (BDH) problem. Additionally, we show that our techniques can be used to build a new signature scheme that is secure under the computational Diffie-Hellman assumption without random oracles.

Identity-based encryption is a very convenient tool to avoid key management. Recipient-privacy is also a major concern nowadays. To combine both, anonymous identity-based encryption has been proposed. This paper extends this notion to adversaries (the authority itself). Accordingly, we compromise the exactness of search trapdoor by mapping at least two search (PEKS).

We prove our scheme to be semantically secure in the Random Oracle (RO) model. The search complexity of our scheme is dependent on the actual number of the cipher containing the queried keyword, rather than the number of all cipher. Finally, we present a generic SPCHS construction from anonymous identity based encryption and collision-free full-identity malleable Identity-Based Key Encapsulation Mechanism (IBKEM) with anonymity.

2. Introduction

Public-key encryption with keyword search (PEKS) is the first keyword searchable encryption based on the probabilistic public key system. It is more convenient to search ciphertexts for multiple users, compared with previous schemes based on a symmetric key system, such as \cite{2,3,4,5}. So far, all of proposed PEKS schemes and their expansions had proved their SS-CKA security. However under keyword guessing attack (KGA), their provable securities rely on an implicit assumption that the size of keyword space must be beyond the polynomial level.

The first efficient and secure method for Identity Based Encryption was put forth by Boneh and Franklin\cite{4}. They proposed a solution using efficiently computable bilinear maps that was shown to be secure in the random oracle model. Since then, there have been schemes shown to be secure without random oracles; the possibility of such a scheme was thought to be an open problem. However, their scheme is too short inefficient to be of practical use.

One of the prominent works to accelerate the search over encrypted keywords in the public-key setting is deterministic encryption introduced by Bellare et al.\cite{2} focus on enabling search over encrypted keywords, such that a ciphertext containing a given keyword can be retrieved in time complexity logarithmic in the total number of all ciphertexts.

3. Identity Base Encryption

Anonymity for public-key encryption schemes has first been introduced by Bellare et al.\cite{3}, under the key privacy security notion, and has been extended to identity-based encryption by Abdalla et al.\cite{1}. In these papers, anonymity meant that even if the adversary chooses a message and two identities (or two public keys), and the challengers encrypts the message with one of the identities (or keys), the adversary cannot guess which one has actually been involved in the computation. This notion is quite strong for public-key encryption, but not that strong in the identity-based setting since it does not capture anonymity with respect to the authority that knows the master secret key, and even chooses the public parameters PK.
3.1 Definition 1 (Identity-Based Encryption).

Setup IBE (1λ). Takes as input a security parameter λ. It outputs the public parameters PK, as well as a master secret key MK. Extract IBE (MK,ID). Takes as input the master secret key MK, and the identity ID of the user. It outputs the user’s decryption key use.

Encrypt IBE (PK,ID,M). Takes as input the public parameter PK, the identity of the recipient, and a message M to be encrypted. It outputs a ciphertext.

Decrypt IBE (use, c). Takes as input the user’s decryption key and a ciphertext c. It outputs the decryption or, if the ciphertext is not valid.

3.2 Definition 2 (Identity-Based Encapsulation)

An IB-KEM scheme is specified by the following algorithms: Takes an input a security parameter λ. It outputs the public parameters PK, as well as a master secret key-MK. Extract IBE (MK,ID). Takes as input the master secret key MK, and the identity ID of the user. It outputs the user’s decryption key use. Encrypt IBE (PK,ID,M). Takes as input the public parameter PK, the identity of the recipient, and a message M to be encrypted. It outputs a ciphertext. Decrypt IBE (use, c). Takes as input the user’s decryption key and a ciphertext c. It outputs the decryption or, if the ciphertext is not valid.

We first review the notion of semantic security for IB-KEM, then we deal with anonymity, and an additional security notion, that we call identity-based non-malleability.

In search Pattern Privacy model[14], the adversary receives two trapdoors and is asked to determine whether the two trapdoors encode the same or different keywords, which are uniformly sampled from the keyword space to ensure that these are sufficiently unpredictable and avoid brute force attacks.

4. Function Privacy For Ibe: An Independent Security Notion

We first remark that Key Unlink ability and Function Privacy security models are essentially different in the way the challenger samples ids: in the former ids are sampled from an adversarial-chosen joint problem space, whereas in the latter model ids may be sampled from an adversarial-chosen joint ability distribution, with(possibly)non-uniform random variables, but also high min-entropy requirements. In the following subsections we provide counters show that function privacy.

The minimal unpredictability requirement of (log λ) bits has only been achieve later in [10]. Schemes in[9] have only been proven secure for highly unpredictable identities with min-entropy of λ+ω (log λ) independent security notions. Meaningful counterexamples follow. For a quick overview, the relation between weak key unlink ability, strong key privacy security notions.

Informally, since f is unknown to the adversary, the adversary on id can be acquired. Therefore, still an enhanced function-private IBE. But, because f is the deterministic, it is trivial to identity with overwhelming probability if two keys have been extracted from the same identity.

4.1 The Security Model

We assume that the message senders possess a valid copy of the receiver’s public key and the
receiver possesses valid copies of the public keys should follow some standard practice, and we skip the discussion in this paper.

Outside attacker: This type of attacker is not assigned with any type of (unencrypted) trapdoor generated under its public key. Curious Type-1 server $S_1$: This type of attacker has been assigned with only message-dependent trapdoors generated under its public key.

Curious Type-2 server $S_2$: This type of attacker has only been assigned with a master trapdoor generated under its public key. Curious hybrid server $S_3$: This type of attacker has been assigned with a master trapdoor and message-dependent trapdoors generated under its public key.

It is clear that a hybrid server is more powerful than others. However, due to the fact that the data owner may employ all three types of servers, it is necessary to consider the maximal level of security against each of them independently.

1. $(P\ Kr, SKr) \xleftarrow{\$} \text{Key Gen}(\lambda)$
2. $(w_0, w_1, \text{state}) \xleftarrow{\$} \text{A}(P\ Kr; s\text{KeyGen}_1, s\text{KeyGen}_2, s\text{KeyGen}_h, \text{TrapGen}_1, \text{TrapGen}_2, \text{Decrypt})$
3. $b \in \mathbb{R} \{0, 1\}, cwb = \text{Encrypt}(wb, P\ Kr)$
4. $b' \xleftarrow{\$} \text{A}(P\ Kr, \text{state}, cwb; s\text{KeyGen}_1, s\text{KeyGen}_2, s\text{KeyGen}_h, \text{TrapGen}_1, \text{TrapGen}_2, \text{Decrypt})$.

Allowed to query any server’s private key through outside attacker can eavesdrop on the transmission of (encrypted) trapdoors, so that we offer it access to both trapdoor generation oracles with its own inputs.

4.2 Forward and Backward Security

Even in the case that a sender gets his local privacy compromised, SPCHS still offers forward security. This means that the existing hidden structure of ciphertexts stays confidential. Since the local privacy only contains the relationship of the new generated ciphertexts. To offer backward security with SPCHS, the sender can initialize a new structure by algorithm structure initialize for the new generated ciphertexts. Because the new structure is independent of the old, the compromised local privacy will not leak the new generated structure.

Search Complexity: All keyword-searchable ciphertexts can be indexed by their parts’ binary bits. Assume that there are in total $n$ ciphertexts from $n_i$ hidden structures.

Experiment: We coded our SPCHS scheme, and tested the cost of algorithm structured search to execute its cryptographic operations for different numbers of matching ciphertexts. We also coded the PEKS scheme [1]. Table 1 shows the system parameters including hardware, software and the chosen elliptic curve. Assume there are in total searchable ciphertexts. PEKS takes about 53.8 seconds search time per keyword, since it must test all ciphertexts for search. Shows the experimental results of SPCHS. It is clear that the time of cost of SPCHS is linear with number of total ciphertexts. Hence, SPCHS is much more efficient than PEKS.

4.3 Notion of Security

We formalize three notions of security with respect to auxiliary inputs, and prove that all three are equivalent. The first is a simulation-based notion, capturing the intuitive meaning of semantic security: whatever can be computed efficiently given a public key, an encryption. We directly describe the security notion for identity-based key encapsulation mechanism, but one can easily derive them for identity-based encryption.

**SEMANTIC SECURITY:** The semantic security formalizes the privacy of the key. The security game, in the strongest security model (i.e. chosen-ciphertext and full-identity attacks) is the following one.

**Setup:** The challenger runs the setup algorithm on input $1^\lambda$ to obtain the public and parameters $pk$, and the master secret key $MK$. It publishes $PK$.

**Find stage:** The adversary $A$ adaptively issues the following queries:

- Extract query on input an ID: The challenger runs the Extract the decryption key for ID, and then decrypts the ciphertext $c$ with this key. It outputs the resulting ephemeral key, for ID, and then decrypts the ciphertext $c$ with this key. It outputs the resulting ephemeral key. A output a target identity ID*, on which on Extract-query has been asked.

**Challenge:** The challenger randomly gets $(K_{0,e}) \xleftarrow{\$} \text{Encaps}_{\text{IBK}} (PK, \text{ID})$ and $(K_{1,e}, c) \xleftarrow{\$} (PK, \text{ID})$. It flips a bit $b$ and outputs $(K_s, c)$. 

![TIME COST OF SPCHS](image)
Guess stage: The adversary can issue the queries as the same find stage, with the restriction that no Extract-query on input ID* and no Decapsulation-query on input can be asked. The adversary finally output its guess for b.

We then define the advantage of A in breaking the Semantic Security of an IB-KEM scheme with its ability in deciding whether it actually received the real ephemeral key associated to c* or a random one. We donate this security notion by IND, which can thereafter be combined with various oracle accesses, in order to define selective/full-identity and chosen plaintext/ciphertext attacks. More formally, we want the advantage below, to be negligible.

5. Security Notion for Keyword Search

Two main security notions have been defined for key exchange protocols. The first is the semantic security of the key, which means that the exchanged key is unknown to anybody other than the players. The second one is unilateral or mutual authentication, which means that either one, or both, of the participants actually know the key. In the following, we focus on the semantic security, also known as AKE Security.

The semantic security of the session key is modeled by an additional query text(U). since we are working in the Real-or-Random scenario, this Test-query can be asked as many times as the adversary A wants, but to fresh instances only. The freshness notion captures the intuitive fact that a session key is not “obviously “known to the adversary. More formally an instance is said to be fresh if it has successfully completed execution and

1. Neither it nor its partner was corrupted before the session started
2. The attack, on this session, was passive.

6. Conclusion

In this paper, we have first introduced two new security notion for identity-based key encapsulation mechanism: the first one is an enhancement of the usual anonymity, the second one formalizes a kind on non-malleability, with respect to the recipient identity.

Then, we proposed the first scheme that is full is full-ID semantically secure against chosen-message attacks, and that achieves our new security notions.

This paper investigated as-fast-as-possible search in PEKS with semantic security. We proposed the concept of SPCHS a variant of PEKS. The new concept allows keyword-searchable ciphertext to be generated with a hidden structure. Given a keyword search trapdoor., the search algorithm of SPCHS can disclose part of this hidden structure for guidance on finding out the ciphertext of the queried keyword.

References