Privacy-preserving Web Search

Myungsun Kim
Department of Mathematical Sciences
Seoul National University
Seoul, Korea
msunkim@snu.ac.kr

Jihye Kim
Department of Electrical Engineering
Kookmin University
Seoul, Korea
jihyek@kookmin.ac.kr

Abstract— **Since there are many sources of potentially identifying information in web search (*e.g.* IP address), we need to ways to hide major clues to the user's identity. Although there have been prior attempts to address this problem, all of them incur linear round complexity in the number of users. In this paper, we construct a constant-round private web search protocol using decomposable encryption, secure in the honest-but-curious model.**

***Keywords*— privacy, decomposable encryption, shuffle, web search.**

I. Introduction

Web search engines help users search for certain data among a huge amount of information in a minimal amount of time. However, these convenient tools also present a privacy threat to the user: web search engines profile their users by storing and analyzing past searches submitted by them. These searches can reveal a lot of information from individual users or the institutions they work for. For example, a certain medical company may use Google to obtain information about a certain chemical ingredient for a new medicine. If a direct competitive company learns this knowledge, it can infer that this ingredient will be used in the new product provided by its competitor. Furthermore, AOL's release in 2006 of 20 million nominally anonymized searches underlined that search queries contain private information [4].

Since Saint-Jean *et al.* in [5] introduced this privacy-preserving web search problem, assuming an honest-but-curious adversary model, an interesting model to solve this problem was proposed by [1]. Briefly, their proposal is for a group of users to first *shuffle* their search words amongst themselves. After the shuffle, each user has someone's search work (but doesn't know whose), and the users then query the search engine with the word obtained. Finally, the parties all broadcast the result to all others. The main technical tool in this model is a protocol for users to mix their inputs. Later, authors in [2] presented an efficient private shuffle protocol secure in a malicious adversary model for this setting, however, with more communication overhead. In particular, both protocols [1, 2] with re-randomizing and re-shuffling incur a round complexity linearly in the number of users. This gives us a question about whether a constant-round private shuffle protocol can be constructed even in the simplest adversary model, i.e., the honest-but-curious adversary model.

Recently, a new encryption primitive was presented in [3]. This primitive defines ElGamal variant over extension fields to support a decomposable property with which multiple ciphertexts can be compressed such that all plaintexts are decomposable after their decryption. We call this primitive *decomposable encryption*. Decomposable encryption in [3] was proposed mainly to improve the bandwidth overhead. Our orthogonal but interesting observation is that decomposable encryption accumulates messages while *hiding* their origin. Namely, linking information between a message and an originator is concealed after ciphtertexts are compressed into a single message. This property is useful and effective for the shuffle protocol where origin information should be private.

In this paper, we construct an efficient constant-round shuffle protocol for privacy-preserving web search. Our protocol is based on decomposable encryption. To the best of our knowledge, even in the honest-bust-curious adversary model, our construction is the first constant-round private shuffle protocol.

II. Prelinimaries

A. Notation

For a natural number n, [n] denotes the set {1, …, n}. If A is a probabilistic polynomial-time machine, we use a $\leftarrow A$ to denote making A produce an output according to its internal randomness. If U is a set then r $\overset{\$}{\leftarrow} U$ is used to denote sampling from the uniform distribution on U. We denote by λ a security parameter. A function $\mathrm{g}: \mathbb{N} \rightarrow \mathbb{R}$ is called negligible if for every positive polynomial $\mathrm{q}(\cdot)$ there is an integer N such that $\mathrm{g(n)} < 1/\mathrm{q(n)}$ for all n > N.

B. Decomposable encryption

A public-key encryption scheme E=(**KeyGen**, **Enc**, **Dec**) consists of the following algorithms: 1) **KeyGen** is a randomized algorithm that takes a security parameter λ as input, and outputs a secret key *sk* and a puble key *pk*; *pk* defines a plaintext space **P** and a ciphtertext space **C**, 2) **Enc** is a randomized algorithm that takes *pk* and a plaintext m $\in \mathbf{P}$ as input, and outputs a ciphertext c $\in \mathbf{C}$, and 3)**Dec** takes *sk* and c $\in \mathbf{C}$ as input, and outputs the plaintext m.

We say that an encryption scheme is *correct* if, for any key pair $(pk, sk) \leftarrow \mathrm{KeyGen}(1^{\lambda})$ and any m $\in \mathbf{P}$, it is the case that m $\leftarrow \mathbf{Dec}_{sk}\left(\mathbf{Enc}_{pk}(\mathrm{m})\right)$.

978-1-4673-1378-0/12/$31.00 ©2012 IEEE

A public-key encryption E is *decomposable* if we can efficiently recover all original messages from a decrypted ciphertext which is obtained by compression of other multiple ciphertexts. Here compression should be efficient. A formal definition in [3] is as follows:

Definition 1. Let T_1 and T_2 be a polynomial time function from $\mathbf{C}^{\mathrm{k}}$ to $\mathbf{C} \cup \{\perp\}$ and a polynomial time function from $\mathbf{P}$ to $\mathbf{P}^{\mathrm{k}} \cup \{\perp\}$ for some k where ⊥ is a distinguished symbol indicating transformation failure. Then, decomposable encryption is given by a tuple of algorithms (**KeyGen**, **Enc**, **Dec,** T_1 , T_2) having the properties below.

1. Easy to compress: For any vector of ciphertext $\mathbf{c} = (\mathrm{c}_1, \dots, \mathrm{c}_\mathrm{k})$, T_1 ($\mathbf{c}$) outputs another ciphtertext $\mathrm{C} \in \mathbf{C}$ or ⊥ where $\mathrm{c}_\mathrm{i} = \mathbf{Enc}_{pk}(\mathrm{m}_\mathrm{i})$**.**
2. Easy to compress: For any plaintext $\mathrm{M} = \mathbf{Dec}_{sk}(T_1$ ($\mathbf{c}$)) $\in \mathbf{P}$ with some vector of ciphtertexts $\mathbf{c}$, $T_2(\mathrm{M})$ outputs a set of messages $\mathbf{m} = \{\mathrm{m}_1, \dots, \mathrm{m}_\mathrm{k}\}$ or ⊥ for some k.
3. Correctness: For any vector of plaintexts $\mathbf{m} = \{\mathrm{m}_1, \dots, \mathrm{m}_\mathrm{k}\}$ be a vector of input messages, and any vector of ciphtertext $\mathbf{c} = (\mathrm{c}_1, \dots, \mathrm{c}_\mathrm{k})$ with $\mathrm{c}_\mathrm{i} = \mathbf{Enc}_{pk}(\mathrm{m}_\mathrm{i})$, it holds that $\{\mathrm{m}_1, \dots, \mathrm{m}_\mathrm{k}\ ) = T_2 \circ \mathbf{Dec} \circ T_1(\mathbf{c})$.

For the detailed definition, refer to [3].

C. Security Experiment

We follow the security experiment for shuffle as in [2]. Given k users, a private shuffle functionality is the n-ary probabilistic function $\mathrm{f}(\mathrm{x}_1, \dots, \mathrm{x}_\mathrm{k}) = (\mathrm{y}_1, \dots, \mathrm{y}_\mathrm{k})$, such that for every i, $\mathrm{y}_\mathrm{i} = \mathrm{x}_{\mu(\mathrm{i})}$ where μ is a random permutation. Intuitively, a shuffle is *privacy-preserving* if an adversary cannot link between the inputs of the protocol and the outputs of the protocol. We formalize security by requiring that an adversary with t corrupted users can output (i, j) where P_i is honest and j= μ(i) with probability that is at most negligibly greater than 1/(k-t).

In the following security experiment **ExSH**:

1. Invoke the adversary $\boldsymbol{A}$ with a security parameter and with parameters t and k (t the number of corrupted parties, and k the overall number of parties).
2. Receive from $\boldsymbol{A}$ a set of t indices I ⊂ [n] designating the corrupted parties (note that |I| = t).
3. Initialize the i-th honest-user oracle with random input w_i and execute the shuffle protocol, where $\boldsymbol{A}$ interacts with the n-t oracles (each oracle runs the specified shuffle protocol as an honest user reponding to the messages it receives from $\boldsymbol{A}$).
4. When it concludes, i.e., it outputs $\{\mathrm{w'}_1, \dots, \mathrm{w'}_\mathrm{k}\}$ where $\mathrm{w}_i = \mathrm{w'}_{\mu(i)}$, the adversary outputs a pair (i, j) for i, j of its choice.

we say that the adversary succeeds in the experiments, in which case the output of the above experiment equals 1, if and only if μ (i) = j.

Definition 2. A protocol μ is a privacy-preserving shuffle if for every probabilistic polynomial time algorithm $\boldsymbol{A}$, every integer k and every 0<t<k, there exists a negligible function negl() such that: Pr[**ExSH**=1] $\leq 1/(\mathrm{n} - \mathrm{t})$ + negl(k).

III. Protocol

We construct a constant-round shuffle protocol based on decomposable encryption in the following:

Input: There are k>2 users P_i with a private search word $\mathrm{w}_\mathrm{i} \in \mathbf{M}$

Output: $\{\mathrm{w}_1, \dots, \mathrm{w}_\mathrm{k}\}$

1. All the users jointly generate a group public/secret key pair pk, sk and corresponding user secret shares for decomposable encryption E
2. Each user P_i broadcasts $\mathrm{c}_\mathrm{i} = \mathbf{Enc}_{pk}(\mathrm{w}_\mathrm{i})$**.**

   Each user P_i receives all $\mathrm{c'}_\mathrm{j}$'s and compute $\mathrm{C} = T_1(\mathrm{c}_1, \dots, \mathrm{c}_\mathrm{k})$.
3. All the users perform group decryption of C to accomplish M=$\mathbf{Dec}_{sk}$(C).

   Each user P_i obtains $\{\mathrm{w'}_1, \dots, \mathrm{w'}_\mathrm{k}\} \leftarrow T_2(\mathrm{M})$ where $\mathrm{w}_\mathrm{i} = \mathrm{w}_{\mu(\mathrm{i})}$

Figure 1. Private shuffle protocol for honest-but-curious adversary

Theorem 1. Assume that the decomposable encryption E is semantically secure. Then, the protocol in Figure 1 is a private shuffle protocol in the honest-but-curious adversary model.

Proof. (sketch) After T_1 function combines all the ciphertexts, all link information between the words and the users is erased. Thus, the adversary can learn the link information only from the ciphertexts, i.e., $\mathrm{c}_1, \dots, \mathrm{c}_\mathrm{k}$. By the semantic security of the decomposable encryption, the advantage of the adversary is no more than random guessing. Given the negligible advantage ϵ over security of the decomposable encryption and the standard hybrid argument, the advantage of the adversary against a private shuffle protocol is: Pr[**ExSH**=1] $\leq 1/(\mathrm{n} - \mathrm{t})$ + kϵ.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2009-0070095). Jihye Kim is the corresponding author for this paper.

References

[1] J. Castellà-Roca, A. Viejo, and J. Herrera-Joancomartí, "Preserving user's privacy in web search engines", Computer Communications 32(13-14), 2009, pp. 1541-1551

[2] Y. Lindell and E. Waisbard, "Private Web Search with Malicious Adversaries", Privacy Enhancing Technologies, 2010, pp. 220-235

[3] Myungsun Kim, Jihye Kim, Jung Hee Cheon, "Compress Multiple Ciphertexts using ElGamal Encryption Schemes", IACR Cryptology ePrint Archive, 2012: 243

[4] M. Barbaro and T.Zeller, "A face is exposed for AOL searcher no. 4417749", New York Times, 9:2008, 2006.

[5] F. Saint-Jean, A. Johnson, D. Boneh, and J. Feigenbaum: Private web search. WPES 2007, pp. 84-90.