Keyword-based Private Information Retrieval 기법의 실용화를 위한 구조적 한계점 분석 연구

박홍근, 정경헌, 권태경

서울대학교

hgpark@mmlab.snu.ac.kr, ghjeong@mmlab.snu.ac.kr, tkkwon@snu.ac.kr

A Study on the Structure Limitations of Keyword-based Private Information Retrieval for Practical Deployment

Hong Geun Park, Gyeong Heon Jeong, Ted "Taekyoung" Kwon

Seoul National Univ.

요 약

현대 인터넷 환경에서는 사용자와 서버 사이에서의 안전한 통신을 위해 TLS 프로토콜이 사용되며, 이를 통해 제 3자로부터 통신 내용 보호가 가능해졌다. 그러나 개인 정보 기반 추천 시스템 및 개인정보 유출 사고 등을 계기로, 서버로부터 자신의 질의 내용을 은닉하고자 하는 요구가 대두되었으며, 이에 따라 사용자의 질의 내용을 숨길 수 있는 Private Information Retrieval (PIR) 기법이 활발하게 연구되고 있다. PIR은 공개 DB 서버의 개수나 질의가 이루어지는 방식에 따라 ITPIR, index-based CPIR, KWPIR로 분류될 수 있으며, 실용화를 위해 다양한 최적화가 진행되었으나, KWPIR의 경우에는 여전히 range query를 지원하기 어렵다는 구조적 한계가 존재한다. 따라서 본 논문에서는 KWPIR의 range query 도입을 어렵게 만드는 keyword 범위 정의의 모호함과 확률적 필터 기반 구조라는 두 가지 근본적인 문제를 지적한다. 이 두 가지 문제는 KWPIR의 실용화를 저해하는 핵심 장애 요소로 작용하고 있으며, 본 논문은 이를 극복하기 위해 range query 친화적인 새로운 구조적 접근 방식의 필요성을 제안한다.

Ⅰ. 서 론

현재의 인터넷 환경에서는 사용자와 서버 사이에서의 안전한 통신을 위해 기밀성(confidentiality), 무결성(integrity), 인증(authenticity)을 보장하는 Transport Layer Secuirty (TLS) 프로토콜이 사용되고 있다. TLS는 제 3자가 사용자와 서버 사이의 통신 내용을 확인할 수 없도록 하여 기본적인 프라이버시 보호를 제공한다. 그러나 사용자의 검색 기록 등의 개인 정보에 기반하여 콘텐츠을 추천하는 알고리즘이 등장함에 따라, 사용자들은 서버로부터도 자신의 정보를 숨기고자 하는 요구를 보이기 시작했다. 또한 몇몇 IT 기업들로부터 사용자의 개인 정보가 유출되는 사고가 발생함에 따라 프라이버시 보호의 중요성이 더욱 부각되고 있다.

이처럼 통신하는 서버를 믿을 수 없는 상황에서도 프라이버시를 보호하기 위해 개발된 기법들이 Oblivious RAM (ORAM)과 Private Information Retrieval (PIR)이다. 두 기법 모두 사용자의 질의 내용을 서버로부터 은닉할 수 있는 기술이지만, ORAM은 사용자의 개인 키로 암호화된 DB 상에서 작동하며, PIR은 개인 키로 암호화될 수 없는 공개 DB 위에서 작동한다[1, 2]. 그러나 ORAM과 PIR 모두 그 중요성에도 불구하고, 높은 계산복잡도 및 공간복잡도로 인해 실용화되기 어렵다는 단점이 있다. 특히 PIR의 경우에는 한 번에 여러 데이터를 받아오기 어렵다는 문제점이 존재한다. 따라서 본 논문에서는 현재 연구되고 있는 PIR의 종류들을 소개하고, 그 중 keyword-based computationally secure PIR (이하 KWPIR) 실용화를 가로막는 현실적인 문제점들을 제시하고자 한다.

Ⅱ. 본론

본 논문에서 KWPIR의 현실적인 문제점을 논의하기에 앞서, KWPIR의 개념에 대해 설명하고자 한다. 우선 PIR은 computationally secure PIR (이하 CPIR)과 information-theoretically secure PIR (이하 ITPIR)로 나누어진다. 전자는 단일 공개 DB 서버 환경을 가정하며, 사용자의 질의를 암호화하여 악의적인 DB 서버가 사용자의 질의에 대한 정보를 알 수 없게 하는 기법으로 주로 LWE-based 동형암호를 사용한다. 반면, ITPIR은 여러 개의 공개 DB 서버에게 질의를 하는 상황을 가정한다. ITPIR에서는 서버들은 사용자의 질의로부터 사용자가 어떤 데이터를 요청하는지를 알 수 없지만, 사용자는 서버들의 응답을 조합함으로써 자신이 원하는 데이터를 획득할 수 있다.

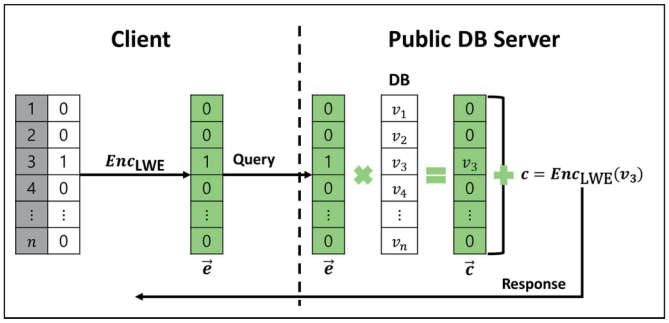

그림 1 사용자가 공개 DB에서 3번째 데이터를 요청했을 때의 index-based CPIR 과정. 초록색 상자는 해당 메시지가 LWE 암호화 되었음을 의미하며, 초록색 연산자는 LWE 암호 연산자를 의미한다.

CPIR은 공개 DB 상의 데이터를 받아오는 방법에 따라 index-based CPIR과 keyword-based CPIR로 다시 나누어진다. Index-based CPIR은 사용자가 데이터를 해당 데이터의 index를 바탕으로 질의하는 CPIR이며, KWPIR은 key-value map으로 이루어진 공개 DB에서 데이터를 해당 데이터의 keyword를 바탕으로 질의하는 CPIR이다. 그림 1은 사용자가

index-based CPIR을 하는 과정을 보여준다. 사용자는 공개 DB 서버에 자신이 질의하고자 하는 데이터를 나타내는 indicator vector를 LWE 암호화한 query vector를 보내고, 공개 DB 서버는 자신의 평문 DB와 query vector를 곱한 값을 응답으로 사용자에게 전송한다. 사용자는 해당 값을 복호화함으로써 자신이 원하는 데이터를 획득할 수 있다.

KWPIR은 주로 index-based CPIR을 기반으로 구현된다. 가장 단순한 구현으로 이진탐색을 통해 keyword가 몇 번째 keyword인지 알아낸 다음 해당 index로 질의를 하는 방법이 있으나 O(log |DB|) 만큼의 질의가 요구되기에 비효율적이라는 단점이 있다. 이를 해결하기 위해 사용자가 질의한 keyword가 DB에 있는지를 확률적 필터(probabilistic filter)를 사용하여 판단하는 기법들이 연구되고 있다[2, 3].

Query format:

SELECT $aggregate_1, aggregate_2, \ldots$
FROM *table*
WHERE *condition*
[GROUP BY $expr_1, expr_2, \ldots$]

그림 2 SQL에서 사용되는 질의 형식[4].

이러한 PIR 기법들은 실제 시스템에 적용하기 위해 여러 최적화를 통해 계산복잡도나 공간복잡도를 줄이는 방식으로 발전되어 왔다[3, 7]. 기존의 PIR들은 한 번의 질의로 한 번의 데이터만을 반환할 수 있었으며, 이는 그림 2의 실제로 널리 사용되는 range query 기반 SQL 명령어들과는 호환되지 않는다는 한계를 지니고 있었다[5]. 다중 공개 서버 환경을 가정하는 ITPIR에서는 Function Secret Sharing (FSS)를 활용하여 공개 DB에 특정 함수 (일부 sql 명령어)들을 수행할 수 있게 함으로써 range query의 구현이 가능해졌다[4]. 한편, index-based CPIR의 경우에는 indicator vector에 복수의 index를 담아 질의하는 batch PIR [6]을 확장하여 range query를 지원하고 있다[5].

그러나 KWPIR은 두 가지 근본적인 문제로 인해 range query를 구현하는 데 어려움이 있다. 첫째, keyword는 index와 달리 범위를 명확히 정의하기 어렵다. 예를 들어, index의 경우 “2번째부터 5번째까지”라는 질의는 구체적으로 2, 3, 4, 5번 데이터를 의미하므로 범위가 명확히 설정된다. 그러나 keyword의 경우, “apple과 banana 사이”의 데이터를 질의한다고 했을 때, 해당 범위에 포함되는 키워드가 무엇인지 명확히 정의하기 어렵다. 또한 이를 공개 DB의 keyword로 한정시킨다고 하더라도, 사용자가 공개 DB의 모든 keyword들을 알고 있어야 한다는 현실적인 상황에서 성립하기 어려운 가정을 필요로 한다는 한계점이 있다.

두 번째 근본적인 원인은 KWPIR의 구조적 특성에 있다. KWPIR은 효율성을 위해 확률적 필터를 사용하며, batch CPIR 과정에서는 복수의 query vector들을 병렬적으로 연결한 query matrix 형태로 질의가 이루어진다[2]. 이러한 구조로 인해, 복수의 인덱스를 하나의 query vector에 포함시키는 방식을 KWPIR에 적용하는 데에는 한계가 존재한다. 즉, KWPIR은 구조적으로 하나의 query vector 내에서의 다중 index 표현을 지원하지 않기 때문에, 기존 CPIR에서 활용되는 batching 기법을 직접적으로 적용하기 어렵다.

Ⅲ. 결론

본 논문에서는 공개 DB 서버에게 자신의 질의 내용을 숨길 수 있는 기법인 PIR에 대해서 소개하고, 특히 KWPIR의 실용화에 초점을 맞추어 논의하였다. PIR은 단일 공개 DB 서버 환경을 가정하는 CPIR과 복수의 공개 DB 서버 환경을 가정하는 ITPIR로 나누어진다. CPIR은 질의 방법에 따라 데이터의 index를 기반으로 질의하는 index-based CPIR과 데이터의 keyword를 기반으로 질의하는 KWPIR로 나누어진다. 이러한 PIR 기법들은 기본적으로 하나의 질의당 하나의 데이터를 받아올 수 있기에, SQL과 같은 range query를 지원하기 어렵다는 구조적 한계점이 존재했다. 이에 따라 ITPIR과 index-based CPIR은 각각 range query를 FSS와 batch PIR을 통해 제한적으로 지원하고 있다. 반면, KWPIR의 경우에는 질의 범위를 정의하기 위해서는 비현실적인 가정이 필요하거나, 확률적 필터를 사용한다는 구조로 인해 batching 기법을 직접적으로 적용하기 어렵다는 한계점이 존재한다. 즉, 현재 연구되고 있는 KWPIR의 2가지 근본적인 문제로 인해 range query의 도입이 지연되고 있으며, 이는 KWPIR의 실용화를 가로막는 핵심 장애 요소로 작용하고 있다. 따라서 KWPIR의 실용화를 위해서는 index-based CPIR 위에 확률적 필터를 결합하는 기존의 방식에서 벗어나, range query 친화적인 새로운 구조적 접근 방식이 요구된다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. RS-2023-00220985).

참 고 문 헌

[1] Goldreich, O., "Towards a theory of software protection and simulation by oblivious RAMs," Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC '87), pp. 182 - 194, May 1987.

[2] Celi, S. and Davidson, A., "Call Me By My Name: Simple, Practical Private Information Retrieval for Keyword Queries," Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS '24), pp. 4107 - 4121, Nov. 2024.

[3] Hao, M., Liu, W., Peng, L., Zhang, C., Wu, P., Zhang, L., Li, H., and Deng, R. H., "Practical Keyword Private Information Retrieval from Key-to-Index Mappings," Cryptology ePrint Archive, Paper 2025/210, 2025. Available at: https://eprint.iacr.org/2025/210

[4] F. Wang, C. Yun, S. Goldwasser, V. Vaikuntanathan, and M. Zaharia, "Splinter: Practical Private Queries on Public Data," in Proceedings of the 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI '17), Boston, MA, USA, Mar. 2017, pp. 299 - 313.

[5] Hayata, J., Schuldt, J. C. N., Hanaoka, G., and Matsuura, K., "On Private Information Retrieval Supporting Range Queries," Proceedings of the 25th European Symposium on Research in Computer Security (ESORICS 2020), Part II, pp. 674 - 694, Sept. 2020.

[6] S. Angel, H. Chen, K. Laine and S. Setty, "PIR with Compressed Queries and Amortized Query Processing," 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2018, pp. 962-979.

[7] Henzinger, A., Hong, M. M., Corrigan-Gibbs, H., Meiklejohn, S., and Vaikuntanathan, V., "One Server for the Price of Two: Simple and Fast Single-Server Private Information Retrieval," Proceedings of the 32nd USENIX Security Symposium (USENIX Security '23), pp. 3889 - 3905, Aug. 2023.