대칭키 방식을 이용한 암호화된 데이터의 키워드 검색에 관한 연구†

이병희, 이윤호, 조석향, 김승주, 원동호*

성균관대학교 정보통신공학부 정보보호연구소

A Study on the Keyword Search on Encrypted Data using Symmetric Key Encryption

Byunghee Lee, Yunho Lee, Seokhyang Cho, Seungjoo Kim, Dongho Won

Information Security Group,
School of Information and Communication Engineering, Sungkyunkwan University

요 약

신뢰할 수 없는 저장매체에 데이터를 안전하게 보관하기 위해서 대부분의 시스템은 데이터를 암호화하는 방식을 사용한다. 암호화된 데이터를 통해서는 원래의 평문에 어떠한 내용이 포함되어 있는지 알 수가 없으며, 해당 데이터의 내용을 열람하기 위해서는 암호화된 데이터 전체를 복호화해야만 한다. 본 논문에서는 암호화된 데이터에 대해 키워드 검색이 가능한 프로토콜을 제안하여, 데이터 전체를 복호화하지 않고 특정 키워드의 포함 여부를 판단할 수 있도록 하였다.

I. 서론

기업체나 개인 사용자가 보관해야할 데이터의 양이 증가함에 따라 데이터 관리에 관한 문제가 대두되었다. 이러한 상황에서 데이터베이스 서비스 (DAS : database as a service)[1]의 등장은 비용적인 측면에서 뿐만 아니라 가용성과 재앙 등과 같은 예기치 못한 상황에서의 데이터 보호 등 여러 측면에서 많은 이득을 제공하였다.

그러나 데이터베이스 서비스의 가장 큰 문제점은 비밀 정보의 안전한 저장이다. DAS는 데이터를 제3자에게 관리하도록 자신의 데이터를 전달해주는 것이기 때문에 개인의 비밀 정보를 보관하기에는 부적합하다. 이러한 문제점을 해결하는 가장 일반적인 방법은 해당 비밀 정보를 암호화하여 저장하는 방법이다. 그러나 단순한 암호화는 기밀성에 관한 문제는 해결되지만, 사용자가 해당 데이터에 관한 정보를 검색하기 위해서 데이터 전체를 자신의 시스템으로 전송하여 데이터의 복호화 과정을 거친 후에야 정보를 얻을 수 있다는 불편함을 초래한다.

본 논문에서는 이러한 문제점들을 해결하기 위한 암호화된 데이터에 대한 키워드 검색 방식을 제안한다. 제안하는 방식을 통해 사용자는 자신의 데이터 전체를 복호화하지 않고도 자신이 선택한 특정 키워드의 포함 여부를 판단함으로써 효율적인 검색이 가능하다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구 기술들에 대해 살펴보고, 3장에서는 본 논문에서 제안하는 방식을 설명하고, 4장에서는 제안하는 방식의 특징을 살펴본다. 마지막으로 5장에서는 논문의 결론에 대해 기술한다.

II. 관련 연구

지난 몇 년간 암호화되어 저장된 데이터의 쿼리 연산에 관한 많은 연구가 진행되어 왔다 [1-3]. 추가적인 인덱스 테이블을 만들어 암호

* 교신저자 : 원동호(dhwon@security.re.kr)

† 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터(ITRC) 육성·지원사업의 연구결과로 수행되었음.

화된 데이터와 함께 저장하는 방식을 제안한 [2, 3]에서는 암호화된 데이터베이스에 대해 선택적 쿼리가 가능하도록 hash 기반의 방식을 제안했다. [4, 5]에서는 쿼리의 조합에 의한 검색이 가능하도록 Privacy homomorphism을 적용하였고, Privacy homomorphism에 대해 수학적 연산(+, -, *, /을 지원하도록 한 기법도 연구되었다[6].

최근에는 공개키 암호화 방식을 사용하여 암호화된 데이터의 검색에 관한 연구 결과가 발표되었다[7-9]. 각각의 데이터와 연관된 secure index data structure는 검색을 요청한 사용자에게 키워드 w에 대한 trapdoor를 제공함으로써 인덱스에서의 키워드 x의 포함 여부를 판단할 수 있도록 해준다. 여기서 인덱스는 키워드 x와 요청자의 공개키를 이용하여 연산이 수행되고, trapdoor에서는 키워드 x와 요청자의 개인키를 이용하여 키워드의 포함 여부에 대한 연산을 수행한다. 물론 이 과정에서 키워드 x가 포함된 문서를 검색하는 것 외에 다른 정보들은 데이터베이스 서버에 노출되지 않는다.

III. 대칭키 암호방식을 이용한 키워드 검색

본 논문에서 제안한 방식은 기본적으로 대칭키 암호방식을 사용하고 있다. 대칭키 암호방식은 공개키 암호방식에 비해 연산 속도가 빠르며 본 논문에서 제안하는 방식은 사용자와 서버 간의 통신이 한 번만 일어나기 때문에 더욱 효율적이라는 장점을 가지고 있다.

3.1 요구사항

데이터가 저장된 데이터베이스 서버(DB 서버)는 신뢰할 수 없는 저장매체이므로 아래와 같은 요구사항이 충족되어야 한다.

- DB 서버 관리자라 하더라도 해당 데이터에 대한 정보는 검색할 수 없다.
- 데이터 소유자와 허가된 검색자 외의 제3자에게는 검색하고자 하는 키워드에 대한 어떠한 정보도 노출되지 않아야 한다.
- 키워드 검색 도중 해당 데이터와 키워드에 대한 정보가 DB 서버에 노출되지 않아야 한다.
- DB 서버와의 통신 도중 공격자가 통신 중인 데이터를 얻을 수 있더라도 해당 암호문에 대한 평문의 정보를 얻을 수 없어야 한다.

3.2 기본동작과정

제안된 방식은 저장 과정과 검색 과정의 2단계로 이루어진다. 각 단계에서 사용되는 파라미터들은 다음과 같다.

[파라미터]

- W_i : 데이터 소유자가 데이터 저장 시 설정한 데이터에 대한 keyword.
- $masterkey$: 데이터 소유자만이 가지고 있는 개인키.
- $PRNG$: 의사난수 발생기.
- $H(\bullet)$: 일방향 해쉬 함수.
- K_i : 데이터 소유자의 masterkey로부터 계산된 키워드 암호화 키. $K_i = H^i(masterkey)$
- C : DB 서버에 저장된 암호화된 키워드.

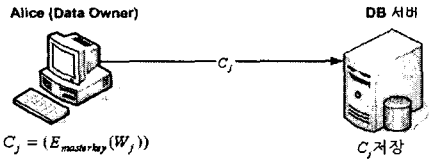
K_i값 중 K_1은 사전에 미리 데이터 소유자와 DB 서버 간에 공유되어지는 값이다.

3.2.1 저장과정

먼저 키워드에 대한 데이터는 이미 DB 서버에 저장되어 있다고 가정한다. 키워드를 저장하고자 하는 데이터 소유자를 Alice라 할 때, Alice는 DB 서버를 신뢰할 수 없기 때문에 자신의 정보를 $masterkey$로 암호화하여 저장한다.

- Alice는 자신의 $masterkey$로 암호화한 각각의 키워드 W_j의 암호문 C_j를 DB 서버로 전송한다.

  $Alice \rightarrow DB$서버 : $C_j = E_{masterkey}(W_j)$



<그림 3-1> 데이터 저장과정

3.2.2 검색과정

- Alice는 선택한 키워드 W를 자신의 $masterkey$로 암호화한다.
- Alice는 임의의 i값을 선택하여 K_i를 생성한다. 생성된 K_i를 키로 사용하여 $masterkey$로 암호화된 키워드 값을 다시 한번 암호화한다.

$$K_i = H^i(masterkey)$$
$$C' = E_{k_i}(E_{masterkey}(W))$$

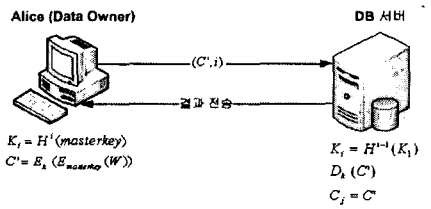
- 생성된 암호화 데이터와 K_i의 인덱스 값인 i를 DB 서버로 전송한다.

$$Alice \rightarrow DB\text{ 서버} : (C', i)$$

- DB 서버는 미리 전송된 K_1과 전송받은 i 값을 이용하여 Alice로부터 받은 C_i를 복호화한다.

$$K_i = H^{i-1}(K_1)$$
$$D_{k_i}(C')$$

- 복호화한 데이터를 저장된 데이터와 비교하여 Alice에게 검색 결과를 알려준다.



[그림 3-2] 데이터 검색 과정

IV. 제안 기법의 확장성 및 안전성 분석

4.1 확장성

제안된 방식은 Alice에 의해 허가된 제3자도 해당 데이터를 검색할 수 있다는 특징이 있다. 제3자를 Bob이라 하면, 검색 과정은 다음과 같다.

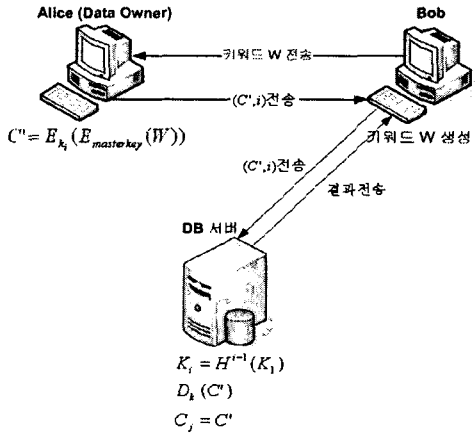
- Bob은 검색하고자 하는 키워드 W를 선택하여 데이터 소유자인 Alice에게 전달한다.
- Alice는 임의의 i를 선택하여 K_i를 계산한다.

$$K_i = H^i(masterkey)$$

- Alice는 자신의 $masterkey$로 암호화된 키워드를 K_i를 키로 사용하여 다시 한번 암호화하여, K_i에 대한 인덱스 값인 i값과 함께 Bob에게 전송한다.

$$Alice \rightarrow Bob : (E_{k_i}(E_{masterkey}(W)), i)$$

- Bob은 Alice에게 전달받은 값을 그대로 DB 서버로 전달한다.
- 값을 전달받은 DB 서버는 3.2.2절의 4번째 단계를 수행한 후 Bob에게 그 결과를 알려준다.



<그림 3-3> 제 3자에 의한 키워드 검색

4.2 안전성

제안된 시스템은 3.1절에서 제시한 요구사항을 아래와 같이 만족한다.

- 데이터의 키워드는 모두 암호화되어 저장되므로 데이터 소유자와 검색자를 제외한 제3자는 물론, DB 서버의 관리자라 할지라도 키워드의 평문에 관한 정보를 얻어낼 수 없다.

- 검색하고자 하는 키워드는 Alice의 *masterkey*로 암호화되어 DB 서버로 전송되기 때문에 Alice로부터 허가를 받은 Bob만이 DB 서버에 저장된 Alice 소유의 정보에 대해 검색할 수 있다.
- Bob으로부터 키워드 검색에 대한 요청이 있을 경우, Alice는 *masterkey*로 암호화된 키워드를 K_i로 다시 암호화하기 때문에, 동일한 키워드를 여러 번 요청하더라도 Bob에게는 다른 값이 전송되기 때문에 선택 평문 공격(chosen-plaintext cryptanalysis)에 안전하다.

V. 결론

데이터를 전문적으로 관리해주는 데이터베이스 서비스(DAS : Database as a Service)의 등장으로 인해 데이터 관리의 측면에서 다양한 이점을 얻을 수 있게 되었다. 그러나 이러한 서비스 업체들은 신뢰할 수 없기 때문에 데이터의 기밀성을 위해 암호 기법을 적용하였지만, 이는 다시 가용성에 대한 문제를 야기하였다.

본 논문에서는 대칭키 암호 방식을 사용하여 외부 저장매체에 저장된 암호화된 데이터에 대한 키워드 검색을 가능케 함으로써 사용자는 암호화된 데이터의 전체 복호화 과정을 거치지 않고도 특정 키워드의 포함 여부를 판단할 수 있는 효율적인 검색 방식을 제안하였다.

[참고문헌]

[1] H. Hacigűműs, B. Iyer, and S. Mehrotra, "Providing database as a service", In Proc. of 18th International Conference on Data Engineering, February 2002.

[2] E. Damiani, S. De Capitani di Vimercati, M. Finetti, S. Paraboschi, P. Samarati, and S. Jajodia, "Implementation of a storage mechanism for untrusted DBMSs", In Proc. of the Second International IEEE Security in Storage Workshop, May 2003.

[3] E. Damiani, S. De Capitani di Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati, "Balancing confidentiality and efficiency in untrusted relational DBMSs", In Proc. of the 10th ACM Conference on Computer and Communication Security, October 2003.

[4] H. Hacigűműs, B. Iyer, and S. Mehrotra, "Efficient Execution of aggregation queries over encrypted relational databases", In Proc. of the 9th International Conference on Database Systems for Advanced Applications, March 2004.

[5] H. Hacigűműs, and S. Mehrotra, "Performance-conscious key management in encrypted databases", In DBSec, 2004.

[6] C. Boyens and O. Gunter, "Using online services in untrusted environments - a privacy- preserving architecture", In Proc. of ECIS '03, June 2003.

[7] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public-key encryption with keyword search", In Proc. of Eurocrypt 2004, May 2004.

[8] E. Goh, "Secure indexes", http://eprint.iacr.org/2003/216.

[9] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log", In Proc. of the 11th Annual Network and Distributed System Security Symposium, February 2004.