情報保護學會論文誌
第 22 卷 第 3 號, 2012. 6

암호화된 데이터베이스 검색 시스템의 보안 요구사항에 대한 통합적 관점에서의 연구

박 현 아,[1†] 이 동 훈,[2] 정 택 영[1‡]
[1]한국과학기술정보연구원, [2]고려대학교

Comprehensive Study on Security and Privacy Requirements for Retrieval System over Encrypted Database

Hyun-A Park,[1†] Dong Hoon Lee,[2] Taik Yeong Chung[1‡]
[1]Korea Institute of Science and Technology Information, [2]Korea University

요 약

지금까지 제안되어져 왔던 대부분의 보안 스킴들이 비록 각종 다른 보안 위협과 공격으로부터 보호하기 위해 그들 자신만의 보안 모델을 연구해 왔다고는 하지만, 이것은 흔히 다음과 같은 문제를 유발할 수 있다 - 어떠한 보안 분석 툴이 어떤 스킴에는 적합하다고 할지라도 다른 스킴에는 부적합 할 수가 있다는 것이다. 이 문제를 설명하기 위해서 본 논문에서는 각 스킴의 보안요구 사항이 어떻게 다를 수 있는지를 Agrawal et al.의 스킴 OPES와 Zdonik et al.의 스킴 FCE를 비교 분석하여 보인다: Agrawal et al.의 스킴 OPES는 Zdonik et al.이 OPES의 안전성을 정형화된 방법으로 안전하지 않다고 반증했기 때문에 현실 상황에 적용 불가능하다고 여겨져 왔다. 하지만, Zdonik et al.의 분석 방법은 객관적인 타당성을 가지지는 않는다. 왜냐하면, Zdonik et al.은 OPES와 그들 자신의 스킴 FCE가 다른 차이점을 가지는 데도 불구하고 그들의 스킴 FCE를 위한 보안 분석 모델(INFO-CPA--DB)로 OPES를 분석하였기 때문이다. 어떤 스킴을 정확하게 분석하고 현실 세계에 적절히 적용하기 위해서는 그 분석 툴은 보편타당한 통합적인 것이 되어야 한다. 따라서 우리 연구의 첫 번째 목적은 모든 암호화된 검색 시스템들을 위한 안전성과 프라이버시 요구 사항에 대한 일반화와 정형화이다. 그리고 나서, 안전한 검색 시스템이 만족해야 할 최소한의 보안 요구 사항과 추가적으로 반드시 고려해야 할 사항을 제언한다. 이것은 암호화 검색 시스템을 바르게 분석함으로써 모든 스킴을 정확하게 평가하여 실제 환경에 올바르게 적용하기 위함이다.

ABSTRACT

Although most proposed security schemes have scrutinized their own security models for protecting different types of threats and attacks, this naturally causes a problem as follows-- if a security analysis tool would fit a certain scheme, it may not be proper to other schemes. In order to address this problem, this paper analyzes how security requirements of each paper could be different by comparing with two schemes: Agrawal et al.'s scheme OPES (Order Preserving Encryp-tion Scheme) and Zdonik et al.'s FCE (Fast Comparison Encryption). Zdonik et al. have formally disproved the security of Agrawal et al.'s scheme OPES. Thereafter, some scholars have wondered whether the OPES can guarantee its applicability in a real world for its insecurity or not. However, the analysis by Zdonik et al. does not have valid objectivity because they used the security model INFO-CPA--DB for their scheme FCE to analyze Agrawal et al.'s scheme OPES, in spite of the differences between two schemes. In order to analyze any scheme correctly and apply it to a real world properly, the analysis tool should be comprehensively standardized. We re-analyze Zdonik et al.'s analysis for OPES and then propose general formalizations of security and privacy for all of the encrypted retrieval systems. Finally, we recommend the minimum level of security requirements under our formal definitions. Additional considerations should be also supplemented in accordance with the conditions of each system.

Keywords: security model, retrieval system, encrypted DB, general formalizations of security and privacy

접수일(2011년 8월 2일), 수정일(1차: 2011년 9월 14일, 2차: 2011년 10월 31일), 게재확정일(2011년 12월 16일)

† 주저자, kokokzi0513@kisti.re.kr
‡ 교신저자, tychung@kisti.re.kr

I. 서 론

대부분의 개인 정보들이 데이터베이스에서 저장되고 프로세싱되기 때문에 민감한 데이터의 효율적이고 안전한 관리는 정보 검색 시스템을 설계하는 데 있어서 필수적인 요소이다. 이런 민감한 데이터를 다양한 종류의 공격으로부터 보호하기 위한 가장 안정적인 솔루션 중의 하나로 암호학적인 방법이나 프라이버시 보호 기술들이 사용되어져 왔다. 지금까지 안전한 검색 시스템에 관한 연구들이 다양한 각도에서 다양한 응용 환경과 시나리오에 대해 많은 학자들에 의해 되어져왔고 각 논문마다 그 환경과 스킴을 위한 안전성이 정의, 증명되어져 있다. 하지만 저자들 자신들에 의해 정의된 안전성과 그에 따른 증명이 과연 객관적인 타당성을 가질 수 있느냐가 문제다. 설령, 그것이 자신들의 스킴과 환경에는 적합할지라도 다른 스킴에까지 적용 가능한 기준이 될 수 있다고 말할 수는 없다.

그런데 [1]에서 Zdonik et al.은 순서를 유지하는 암호화 기법(OPES, order preserving encryption scheme)은 어떠한 것도 안전하지 않다고 주장하면서, [1]의 저자들은 그들의 스킴 FCE(fast comparison encrytion)의 안전성 증명을 위해 정의한 게임 모델 INFO-CPA-DB를 사용하여 [2]의 스킴 OPES가 안전하지 않음을 증명하였다. 여기서 우리는 Zdonik et al.이 주의깊게 살피지 못하고 그냥 간과해 버린 몇 가지 실책을 발견할 수 있다. OPES의 응용 환경은 FCE와 다르고 그 암호화 방법과 위협 모델, 공격 유형 등의 차이점 때문에 그들의 게임 모델 INFO-CPA-DB로 OPES의 안전성을 증명하는 것은 타당하지 못하다는 것이다. [2]의 스킴 OPES의 성능이 효율적이라 할지라도 이런 잘못된 방법으로 안전하지 않다고 [1]에서 증명되었기 때문에 실용화는 불가능하다고 여겨져 온 것이 사실이다. 따라서 숙고를 통해 얻어진 연구 결과가 잘못된 분석으로 인해서 실제 환경에 적용될 기회를 잃지 않도록 하는 것이 우리가 이 논문을 쓰게 된 동기이다.

이 논문에서는 우선 [1]에서 행해진 OPES 분석에 대한 옳고 나쁨을 다시 살펴 본다. 그리고 암호화된 검색 시스템에 관한 우리의 연구에 기반하여 모든 종류의 암호화 데이터베이스 검색 시스템에 대한 시큐리티와 프라이버시 요구 사항을 도출하고, 공격자 유형에 따른 시큐리티와 프라이버시 요구 사항을 일반화 및 정형화하여 정의한다. 그 정의에 기초하여 우리는 안전한 검색 시스템이 갖춰야 할 최소한의 보안 요구사항을 제안한다.

이 논문은 암호화된 데이터의 검색 시스템을 실제 환경에 올바르게 적용할 수 있도록 하기 위해서, 바른 분석을 통하여 모든 스킴을 정확하게 평가할 수 있게 한다.

1.1 관련연구

민감한 정보를 저장하고 있는 검색 시스템(secure database)에 관한 연구는 다양한 각도에서 오래 전부터 연구되어져왔다. 2002년 [3]에서 Agrawal et al.은 DB 프라이버시 원칙과 그것을 만족하는 데이터베이스 모델, ‘strawman’을 제안하였다. 이 논문은 암호화된 데이터 검색 시스템 연구에 있어서 기본 개념을 정립한 것으로, “나는 환자가 나에게 알려준 모든 것에 대하여 비밀을 지키겠노라”는 히포크라테스 선서 내용을 만족하는 데이터베이스, 즉, 데이터베이스에 저장되어 있는 정보가 타인에게 누설되어 개인의 프라이버시가 침해되는 일이 없도록 하자는 것이다. 최근까지도 이런 기본 정신에 입각하여 설계된 많은 연구들이 나오고 있다.

이 분야에 있어서 다른 중요한 연구 중 하나는 [9]에서 제안한 DAS (Database As a Service) 모델이다. 이 모델의 특징은 기존의 연구들과는 달리 서버가 신뢰적인 존재가 아니다. 사용자와 클라이언트, 서버로 구성되는 이 모델에서는 클라이언트가 일종의 신뢰기관 역할을 한다. 클라이언트는 모든 비밀키와 그와 관련된 정보를 알고 있으나 저장량이나 계산 능력이 크지는 못하다. 그래서 사용자가 클라이언트에게 평문으로 자신이 알고자 하는 정보를 질의하면, 클라이언트는 그 질의 내용을 암호화하여 서버에게 다시 질의하게 된다. 비록 비밀키와 그에 관한 정보는 어떠한 것도 모르지만 저장량과 계산 능력이 엄청난 서버는 모든 암호화된 정보를 저장하고 있다가 클라이언트로부터 질의를 받으면 그에 대한 검색을 주어진 프로토콜대로 수행하게 된다. 서버는 암호화된 데이터들의 비밀키를 모르기 때문에 복호화를 수행할 수가 없으며 단지 결과를 클라이언트에게 전송하면 클라이언트가 복호화를 수행하고 평문화된 결과값을 사용자에게 전송한다. 다시 말해서, 비신뢰적인 서버에서 복호화가 절대 일어나지 않는 것이 이 모델의 특징이다. 이것은 서버를 일종의 내부 공격자로 간주함을 의미하는데, 최근 서버 매니저에 의해 일어나는 많은 정보 유출 사례를 살펴 보자면, 현실적으로 상당히 적합한 모델이

라고 할 수 있으며, [9] 이후부터는 많은 연구가 주로 DAS 모델을 기반으로 하고 있다 [5,6,7,27].

다음은 이런 암호화된 데이터에 관한 SQL 쿼리에 관한 연구들이 있다. 그 대표적인 것으로서 [2]의 OPE (order preserving encryption)과 [33]의 Structure Preserving Ddatabase Encryption, privacy homomorphism 성질을 이용한 [8], 그리고 키워드 인덱스 검색 스킴의 연장선에서 제안한 [13], 그리고 이러한 기본 스킴들을 이용하여 보다 더 효율적인 스킴들을 제안한 다른 많은 연구들이 있다 [27,28].

또 다른 암호화된 검색 시스템에 관한 주요한 연구 분야의 한 갈래로서 Song et al. [30]에 의해 처음으로 제안된 키워드 인덱스 검색 시스템에 관한 연구가 있다. 이 분야의 특징은 서버가 비신뢰적인 내부 공격자로, 질의어 및 인덱스, 저장된 정보 모두 암호화 되어져야 하며 서버단에서의 복호화가 없고, 증명 가능한 안전성에 기반한 안전성 위주의 연구 분야라고 할 수 있다. Song et al. 이 대칭키 기반의 $\oplus$ (Exclusive OR) 연산을 이용하여 문서 전체를 스캐닝하는 검색 알고리즘을 제안하면서 키워드를 암호화하여 인덱스로 이용하는 기법에 대해 언급하였다. 그 이후로 이 분야에 대한 연구는 문서 전체가 아닌, 문서를 대표하는 주요 키워드를 추출하여 그것을 각자의 암호화 알고리즘을 이용하여 인덱스를 생성하고, 그 인덱스를 검색하는 스킴에 대한 연구가 이루어졌다. 대칭키 기반의 주요 연구로 [15,24]이 있고, Boneh et al. 에 의해 공개키 기반의 검색 시스템에 관한 연구가 [14]에서 처음으로 소개된 이후, 많은 연구가 공개키 시스템 또는 그것을 기반으로 하여 페어링 함수를 이용한 검색 시스템이 제안되었다. 그리고, Golle et al. [25]에 의해 동시적 검색 (Conjunctive Search)이 처음으로 제안되어진 후, 페어링 함수를 이용한 동시적 검색에 관한 스킴 역시 많이 연구되었다 [4,14]. 그리고 [26]에서 Park et al.에 의해 다이나믹한 그룹 환경을 위한 그룹키 기반의 검색 시스템이 제안된 이후 [34]와 같은 그룹키 환경에서 Conjunctive Search까지 가능하게 하는 스킴들도 연구되고 있으며, [11]과 같이 공개키 기반의 검색 시스템에서 추측 공격(guessing attack)에 대한 취약성을 증명한 공격 논문도 있다.

이것과는 조금 다른 각도에서 Zerr et al. 은 Top-K 검색 알고리즘을 발표했는데, 이것은 exact matching이 아닌 ranked 키워드 검색에 관한 것이다 [35]. Wang et al.은 RSSE(ranked searchable symmetric encryption)의 정의를 제안하고 OPSE(order-preserving symmetric encryption)를 사용하였다 [32]. 그들은 암호화된 클라우드 데이터에 관한 효율적이지만 안전한 ranked 키워드 검색을 고려했다. Cao et al.은 처음으로 암호화된 클라우드 데이터에 대한 멀티 키워드 ranked 검색(multi-keyword ranked search over encrypted cloud data(MRSE))을 연구하고, 안전한 클라우드 데이터 사용 시스템을 위한 프라이버시 요구 사항을 마련하였다[16]. 최근 논문일 경우, 주로 그 응용 환경이 현대 및 차세대 컴퓨팅 환경인 클라우드 서비스의 데이터센터인 경우가 많고 [16, 28, 32], 이 데이터센터의 서버관리자의 특성상 신뢰적인 존재일 수가 없다. 그래서 논문들은 주로 이런 클라우드 서비스 환경 DaaS(Datacenter as a Service)에서 데이터베이스 모델 DAS (Database as Service) 모델을 주로 따르고 있다.

암호화된 검색 시스템에 관한 연구는 굉장히 다양한 각도와 응용 환경에서 연구되어져 왔으며, 그에 대한 안전성 역시 그 응용 환경과 스킴에 적합하게 고려되고 증명되어져 왔다. 하지만 그들이 제안한 스킴의 안전성을 주장하기 위해 각 연구에서 제시되어진 안전성이 과연 실지로 타당한 것인지, 혹시 편협한 관점으로 치우쳐진 것은 아닌지에 관한 의문을 갖게 된다. 물론 각기 다른 환경에서 모든 기준이 똑같이 적용되어져야 한다는 것은 아니다. 각각의 독특한 스킴이라 하더라도 검색 시스템으로서 기본적으로 갖추어야 될 보안적 요소와 프라이버시의 기준은 있을 것이며, 본 저자들은 이 논문에서 바로 그것의 모델을 마련하고자 하는 것이다.

현재까지의 보안 (security) 및 프라이버시 모델에 관한 연구를 살펴보자. 데이터베이스 보안에 관해서는 많은 연구들이 있어왔다. 1988년 Denning의 [22]부터 [36]의 Web DB 보안 모델 등이 그것이다. 그리고 프라이버시 모델에 관한 연구로는 전자 상거래나 웹 서비스에 있어서 프라이버시 정책 요구사항에 관한 논문들이 다수 존재하고, [31]과 같은 RFID에 관한 프라이버시 모델이나 [21]에서 제안한 통계학적 데이터베이스에서 프라이버시 수준을 수량화하는 모델 등이 있다. 이런 연구들 모두는 대체적으로 각자의 환경에서 필요한 요구사항들만을 정리하여 그것을 기반으로 스킴을 디자인하고 그 요구사항들을 만족하는지를 증명한 것이지 검색 시스템을 위한 보안이

나 프라이버시 모델을 공식화한 것들이 아니다.

이 논문에서는 암호화된 데이터 검색 시스템에서의 보안 및 프라이버시 모델을 정의하고 다른 모델에 적용시켜 보도록 한다.

II. OPES 분석의 문제점

2004년에 Agrawal et al.은 그들의 새로운 스킴 OPES (order preserving encryption scheme)을 제안하였다. OPES는 일반 검색 시스템과 비슷한 성능을 가졌기 때문에 매우 효율적이고 실용적이라고 여겨졌다. 그것은 오직 쿼리를 암호화하고 결과값들을 복호화하기만 하면 되기 때문이다. 그런데, Zdonik et al.은 그들 자신들의 스킴(FCE)을 위한 게임 모델 INFO-CPA-DB를 사용하여 OPES의 안전성을 반증하였다. 그 이후로 사람들은 OPES는 안전하지 않기 때문에 실제 환경에는 적용 불가능하다고 생각하게 되었다. 하지만, 우리는 Zdonik et al.의 분석에서 몇 가지 석연치 않은 점을 발견한다. 우선, 그 두 스킴들 사이의 차이점을 보임으로써 Zdonik et al.의 OPES에 관한 분석의 옳은 점과 잘못된 점을 알아보도록 한다.

2.1 DB의 암복호화 방법의 차이점

[1]에서, Zdonik et al.은 "OPES 스킴, 즉 순서를 보존하는 스킴은 어떠한 것이라도 그것의 칼럼들(열들)의 부분집합이 암호화되는 일상적인 사용 시나리오 환경(Zdonik et al.이 가정한 모델) 하에서는 안전하지 않다고 말했다. 현재 상용 제품의 일부는 이 사용 시나리오에 속하지만 모든 사용 모델들이 단지 칼럼의 부분집합만을 암호화하는 것은 아니다. 그것은 각 어플리케이션의 환경과 시큐리티 목적에 따라 다르다. OPES의 목적은 접근제어로부터 우회하고 암호화된 값(따라서, ciphertext only attack이라 불린다)에 접근하는 공격자에 대한 안전성이다. OPES는 모든 disk-resident 데이터를 암호화한다. 테이블과 컬럼 네임과 같은 스키마 정보, 컬럼 통계정보 같은 메터데이터, 로그를 복구하는데 씌여진 값들도 역시 공격자가 이러한 정보를 이용해서 데이터 분포를 추측하는 것을 막기 위해서 암호화해야 하며, 스토리지 시스템에서 복호화 과정이 없다. 반면, Zdonik et al.의 스킴 FCE는 데이터의 단지 일부만을 암호화하고 효율성을 위해 스토리지 시스템에서의 복호화 과정을 가진다. 그러나 사실상 스토리지 시스템에서의 복호화 과정이라는 것은 그 시스템에 접근하는 공격자에 대한 잠재적인 위협을 가진다. 우리가 스토리지 시스템의 데이터를 암호화하는 이유는 접근제어를 피할 수 있는 공격자에 대하여 최종 방어선으로서 공격자를 막아내기 위해서이다. 만약 시스템이 접근 제어를 완벽하게 보장할 수만 있고 서버 매니저의 신뢰도가 보장된다면, 굳이 암호화로 인해 추가 비용을 낭비할 필요는 없다.

2.2 다른 공격 환경: 선택평문공격 vs. 암호문단독공격

비록 Zdonik et al.은 자신들의 위협 모델과 보호의 목적은 OPES에 의해 고려되었다고 말하지만, 앞서 살펴본 바와 같이 두 스킴들은 중요한 시큐리티 측면에서 근본적으로 다르다. 때문에, FCE의 시큐리티 증명 툴에 의한 OPES에 관한 분석이나 FCE 환경하에서의 OPES에 대한 잠재적인 공격 시나리오는 적절하지 않다.

그러나, Zdonik et al.은 그들의 시큐리티 도구 INFO-CPA-DB를 가지고 OPES의 안전성을 반증했다. 시큐리티 게임 모델 INFO-CPA-DB는 선택평문 공격(CPA, chosen plaintext attack)을 위한 것이지, OPES의 위협 모델인 암호문 단독 공격(cipher only attack)을 위한 것이 아니다. 이것은 랜덤 오라클 모델 하에서 CPA에 대해 안전한 스킴을 스탠다드 모델하의 CCA에 대해 안전한 시큐리티 모델로 분석한 경우와 비슷한 경우로, 그 분석 결과는 당연히 '안전하지 않다'이다. 만약 FCE 역시 다른 시큐리티 모델로 분석한다면, 그 안전성 역시 장담할 수 없다. INFO-CPA-DB의 증명 방법과 관련하여, CPA에 대한 증명 과정은 그 데이터의 원래의 분포에 대한 정보를 노출한다. 참가자 A(공격자)가 0~2l-1 사이의 2l 개의 평문값을 선택하고 그 값들에 해당하는 2l 개의 암호문 값을 오름차순으로 생성한다. 참가자 G(암호시스템의 보호자)는 암호문이 주어지면, 시뮬레이션 스크립트를 생성하고 랜덤 메시지 값을 생각해내야 한다. 순서를 보존하는 특성과 평문 도메인의 고정된 크기(2l) 때문에, 참가자 G는 해당 암호문에 대한 각 랜덤 메시지 값을 명백히 결정할 수 있다. 0~2l-1 사이의 2l 개의 정수는 이 도메인의 분포는 'uniform' 분포이로 각 버킷(bucket)의 넓이는 1이다라는 것을 의미한다. 그러나 OPES의 안전성은 어떠한 분포에 위치한 평문값은 사용자가 조작한 타겟

분포에 매핑되어진다는 사실에 기초한다. 이때 각 버킷 넓이를 다양하게 하고 전체 데이터베이스를 암호화하기 때문에 평문 도메인의 원래 분포에 관한 정보는 노출되지 않는다. Agrawal et al.은 그들의 시큐리티 목적을 실험적인 증명을 통하여 성취하였음을 그들의 논문에서 보여주었다. 따라서, Zdonik et al.이 OPES 반증에 대한 타당성을 보이려면, OPES의 실험적 증명의 셋팅하에서 반증에 대한 결과나 모순을 보여야 함이 이치에 맞다.

2.3 유추공격 가능성

Zdonik et al.은 OPES의 유추 공격의 잠재적인 위험성과 시큐리티 척도기 'percentile exposure'의 부적절성을 지적하였다 - 그들은 공격자가 칼럼에서 하나 이상의 평문값을 알게 되면, 순서를 보존하는 스킴들은 유추 공격에 취약하다고 말한다. 그러나 이것은 FCE의 응용 환경 하에서만 가능한 것이다. FCE는 단지 칼럼의 부분집합만을 암호할 뿐이며 스토리지 시스템 상에서의 복호화 과정이 존재한다. 그렇기 때문에 공격자가 평문들을 쉽게 알아낼 수 있지만, OPES는 그렇게 하지 못하도록 설계되어져 있다. 즉, OPES에서 공격자는 어떠한 평문도 쉽게 알 수 없다. 왜냐하면, 모든 disk-resident 데이터와 스키마 정보, 그리고 공격자가 데이터 분포를 추측할 수 있는 모든 정보들은 다 암호화가 되어 있기 때문이다. 뿐만 아니라, OPES의 암호화 방법 역시 평범하지 않다. OPES는 원래의 분포를 사용자가 조작한 타겟 분포로 변화시킨다. 원래의 분포와 타겟 분포에 대한 각 도메인은 다른 사이즈의 버킷들로 각각 쪼개진다. 이러한 다른 넓이를 가진 각 버킷들과 각 버킷의 바운더리의 값들은 암호화 키로서의 기능을 하게 된다. 때문에 비록 공격자가 모든 암호화 값들을 가지고 있다고 할지라도 원래의 평문 데이터베이스 테이블을 유추할 수는 없는 것이다.

2.4 서버단에서의 복호화 가능성

Zdonik et al.은 OPES의 데이터와 그 데이터들의 분포에 관한 사전 정보를 공격자는 가지고 있지 않다는 가정을 타당하지 못하다고 하였다. 실제 환경에서는 민감한 데이터들의 칼럼을 많은 응용 환경들이 가지고 있으며, 그 데이터들의 분포는 공격자에 의해서 쉽게 추측 가능하거나 이미 잘 알려져 있다고 하였다. 뿐만 아니라, OPES는 비신뢰적인 서버 환경에서 사용될 수 있는데, 이런 환경에서는 복호화 과정이 선택적 상황이 아니기 때문에 OPES의 가정하에서는 사용되어질 수 없다는 것이다.

그러나, 일반적으로 비신뢰적인 서버 환경하에서는 어떠한 복호화 과정도 있어서는 안되는 것으로 우리는 알고 있다. 예로, DAS(database as a service) 모델에서[5,6,7,9]는 오직 클라이언트만이 결과를 복호화할 수 있다. 그 이유는 오직 클라이언트만이 비밀키값을 알고 있고, 비신뢰적인 서버는 비밀키값을 알 수 없으며 오직 암호화된 데이터에 대한 검색 과정만을 수행할 뿐이다. 앞서 언급한 바와 같이 데이터를 암호화 시키는 이유는 접근제어를 우회하는 공격자로부터 데이터를 보호하기 위함이다. 비신뢰적인 서버 관리자는 내부 공격자로 간주될 수 있다. 사실, 서버 매니저에 의한 정보의 불법적인 사용은 프라이버시와 관련한 심각한 사회적인 문제[1]를 야기시켜왔다. "Cpmputer Crime and Security Suervey"에 의하면[37], 공격의 45%는 내부 공격자들에 의한 것이라고 한다. 최근, 클라우드 컴퓨팅 서비스가 정보 통신 기술에 있어서 급진적인 전환점을 마련하고 있다. 많은 사람들과 기업들이 스토리지 서비스로 DaaS (datacenter as a service)를 사용할 것으로 기대되는데, 여기서 DAS 모델이 암호화를 이용하여 사용자들의 민감한 정보를 보호할 적절한 데이터베이스 모델이라고 여겨진다. 그렇기 때문에 OPES의 스토리지 시스템에서 복호화 과정이 없는 것이 FCE 환경의 가정보다 더 합리적이다.

III. 연구 범위

암호화된 검색 시스템에서의 통합적인 안전성 평가 툴을 만들기 위해 다음과 같이 연구 범위를 한정한다.

3.1 검색 시스템의 분류

정보보호 측면에서 볼 때, 안전한 검색시스템은 크게 두 가지로 분류되어질 수 있다. 하나는 공개적인 데이터베이스 (publishing DB) 이고 또 다른 하나는 비공개적인 데이터베이스(unpublished DB)이다.

1) 예로 2008년 '하나로 텔레콤'과 'GS 칼텍스' 사건을 들 수 있다.

3.1.1 공개적인 데이터베이스(publishing DB)

연구나 통계학적 분석을 위해 사용되는 병원의 임상 기록같은 통계학적 데이터베이스 (statistical database)에 대한 것으로, 이것은 정보 내용 자체가 민감한 사항이라기 보다는 정보의 주체가 누구인가가 더 민감한 사항이다. 왜냐하면, 정보의 내용은 통계학적 연구 분석을 위해 사용되어야 할 데이터로 공개되어야하기 때문이다. 이러한 환경에서 중요한 문제는 정보채집자(data-miner)나 다른 어떠한 사람이 공개된 데이터베이스를 보고 그 정보의 주체가 누구인지, 혹은 정보 주체에 관한 연구 분석의 목적 외의 다른 어떠한 정보라도 알아낼 수 있으면 안 된다는 것이다.

이를 위해 '데이터마이닝에서의 프라이버시 보호' (PPDM, privacy preserving in datamining)라는 연구 분야에서 데이터마이닝을 가능하게 하는 동시에 정보 주체들의 프라이버시도 보호하게 하는 연구를 하고 있다. 연구 분석을 위한 데이터들은 많은 연산과 복잡한 SQL 질의 과정을 거쳐야 하기 때문에, 이 분야에서 사용되는 기술은 암호화 기법보다는 k-anonimity 혹은 l-diversity 와 같은 익명화 기법이나 perturbation, swapping 등과 같은 방법들이 사용된다.

3.1.2 비공개적인 데이터베이스(unpublished DB)

누구의 정보이냐도 중요한 문제이지만, 그 내용 자체가 더욱 더 민감한 정보이기 때문에 데이터 자체를 암호화하는 경우이다. 이것은 파일 시스템일 수도 있고, 데이터베이스 시스템 (DBMS, database management system)일 수도 있으며, 데이터 전부를 암호화 할 수도 있고 긴밀한 정보 일부만을 암호화 할 수도 있다. 이 논문에서는 이러한 민감한 정보를 암호화한 검색 시스템에 관한 보안과 프라이버시 문제에 집중한다. 이러한 암호화된 검색 시스템은 다시 두 가지 경우로 나눌 수 있다. Web DB(분산 환경) 검색과 DBMS에서의 검색(self-managed DB search, service-provider-managed DB search)이다.

3.1 공격자 유형

다음은 전통적인 공격자의 유형으로 외부 공격자와 내부 공격자를 설명한다. 우리 시스템은 외부 공격자 뿐만 아니라 45% 이상의 내부 공격자 [37] 모두를 방어한다.

1. 외부 공격자 (outside attacker): 서버에 저장된 데이터에 대한 접근 권한이 없거나 제한적이다. 따라서, 허가받지 않은 데이터에 대한 불법적인 접근을 시도한다.
2. 내부 공격자 (inside attacker): 이 논문에서 내부 공격자는 접근 제어의 통제를 벗어난 모든 공격자를 통칭한다. 불법적인 접근 시도에 성공한 외부 공격자나 서버 매니저처럼 데이터에 대한 접근이 원래부터 자유로운 경우를 말한다.

IV. SRS(secure retrieval system) 알고리즘

다음은 SRS를 구성하기 위한 기본적인 셋팅인 필수적인 알고리즘을 정의한다.

Definition 1. 안전한 검색 시스템 (SRS, Secure Retrieval System)은 크게 다음의 7가지 알고리즘으로 구성된다.

1. *SetUpSys*(1^s)

시큐리티 파라메터 (security parameter) s를 입력 변수로 하여 시스템 파라메터 (system parameter) λ를 출력한다. λ는 암호화된 검색 시스템을 셋업하기 위한 구성 요소들로서, 시스템 크기, 암/복호화 알고리즘과 거기에 사용되는 함수 및 파라메터의 크기 등을 결정한다.

2. *KeyGen*(λ)

시스템 파라메터 λ를 입력변수로 받아 문서 암호 및 인덱스 생성시 필요한 비밀키 집합 K를 생성한다.

3. *SetUpEncDB*(K, D)

비밀키 집합 K를 입력 변수로 하여 데이터 D를 암호화한 ($E_K(D)$)와 그것의 인덱스 I_D를 생성한다. 생성되어진 암호화된 데이터와 인덱스들은 각 응용 환경에 적합한 서버 S에 저장된다.

4. *AuthUser*(PI_U)

사용자에 관한 정보 PI_U 를 입력값으로 하여 검증 테스트를 만족하면 '1'을 출력하고, 그렇지 않으면 '0'을 출력한다.

5. *SendQuery(PI_U, W, k)*

질의하고자 하는 검색어 W와 비밀키 k를 입력변수로 하여 서버에게 검색 능력을 부여하기 위한 암호화된 질의어 Q_w를 출력한다.

6. *SearchDB(Q_w)*

암호화된 질의어 Q_w를 받아서 주어진 프로토콜에 의해 데이터베이스를 검색한다. 검색 조건을 만족하면 '1'을 출력하고, 그렇지 않으면 '0'을 출력한다.

7. *Decrypt(R_c, k)*

'1'을 출력한 암호화된 데이터 집합 R_c와 비밀키 k를 입력받아 복호화하여 평문 결과 R_p를 출력한다.

V. 외부 공격자에 대한 접근 제어

이 장에서부터 우리는 암호화된 검색 시스템을 공격 유형별로 나누어 그것을 방어하는 시스템을 구성한다. 그 공격 유형으로는 외부공격자, 내부 및 외부공격자, 내부공격자를 들 수 있고, 먼저 외부 공격자에 대한 접근제어 시스템을 살펴보기로 한다.

민감한 데이터의 불법적인 접근을 제어하기 위해서 검색 시스템에 저장된 데이터의 흐름 및 그 주체 관계를 살펴 볼 필요가 있다. 안전한 검색 시스템은 그 데이터의 소유자(O)/저장소(S)/검색자(U)에 따라 다음과 같이 크게 3가지로 분류되어질 수 있다. [29]의 분류를 기반으로 한다.

- *OSO* 모델 : 데이터 소유자 O가 자신의 정보를 암호화하여 서버 S에 저장하여 자신의 정보를 자기가 필요시 검색하는 경우이다. 데이터 소유자와 검색자가 동일한 경우로, 키워드 인덱스 검색 시스템 연구 분야의 절반 이상이 이 모델을 따른다.
- *SSU* 모델 : 데이터 소유자, 즉 데이터 공급자(data supplier)가 그들의 데이터를 그들의 서버 S에 저장하고 사용자들에게 그 데이터를 제공하는 것이다. 즉, 데이터 소유자와 서버가 동일한 경우로, 이 때 서버는 내부 공격자가 아니다. 이 모델에서의 안전성 및 프라이버시는 사용자의 선호도(preference)로, 서버는 사용자가 어떠한 데이터를 이용했는지 알아서는 안 되며, 단지 몇 개의 데이터가 이용되었는지 만을 알 수 있다. 이 모델의 응용 환경으로는 DVD 혹은 음악 파일 같은 웹 서비스 프로바이더를 들 수 있다.
- *OSU* 모델 : 데이터 소유자 O가 암호화한 데이터를 서버 S에 저장하고 다른 사용자 U가 그 데이터를 검색하는 경우로 사용자, 서버, 검색자가 모두 다르다. 이 모델의 응용 환경으로는 민감한 데이터를 다른 사용자가 공유하는 웹 DB 검색 [8,10]이나 데일리라이프서비스의 개인화 DB의 데이터 공유 기능 [17], 또는 키워드 인덱스 검색 시스템의 이메일 서버로 발신자가 수신자의 공개키로 메일을 암호화하여 메일 서버에 저장하는 경우를 들 수 있다 [14].

이러한 데이터의 소유자 및 그 이용 흐름의 분류에 기반하여 분석해 볼 때, 암호화 검색 시스템에 불법적인 접근을 막기 위한 방법으로 2가지 기술을 효과적으로 사용할 수 있다; 인증과 데이터에 대한 암호화 키를 사용한다.

개체 인증에는 크게 약한 인증과 강한 인증의 두 가지 방법이 있다. 약한 인증(WA, weak authentication)은 6-10개 이상의 숫자나 철자의 조합으로 이루어진 전통적인 패스워드 스킴을 일컫는 반면, 강한 인증(SA, strong authentication)은 다인자 인증법 (multi-factor authentication)의 하나로 다음의 세 가지 요소 중 최소한 두 가지를 동시에 만족하여야 한다. 어떤 계정에 로긴할 때 1) 그 사람이 알고 있는 것 2) 그 사람이 소유한 것, 그리고 3) 그 사람이 어떠한 사람이라는 것에 관한 것이다 [18]

Definition 2. 개체 인증 (Entity Authentication). 한 참여자가 다른 참여자를, 실제로 참여한 그 참여자의 아이덴티티(identity)로, 확신하는 과정을 말한다 [18].

Definition 3. 약한 또는 강한 인증 스킴. *WA* 또는 *SA*의 검증 테스트 AT_W 또는 AT_S는 사용자 U가 정당한 사용자로서 그 테스트를 통과하면 '1'을, 그렇지 않으면 '0'을 출력하고, 'AT_W 또는 $AT_S(U)=1$ 또는 0'으로 표기한다.

암호화키 역시 데이터에 대한 불법적인 접근을 막을 수 있다. 그 솔루션은 다음과 같은 두 가지 경우로 생각되어질 수 있다.

5.1 복호화 주체가 키를 가지고 있는 경우

OSO 모델과 *OSU* 모델의 이메일 서버에 해당하

는 경우로 서버가 검색 후 결과물로 전송한 데이터는 사용자의 암호화키(이메일의 경우: 공개키)로 암호화 되어 있다. 따라서 정당한 사용자가 아니고서는 그 결과를 복호화 할 수 없다, 단지 모든 검색 시스템은 사용자가 로그인할 때 정당한 사용자로서 개체 인증(entity authentication)을 하도록 요구하고 있다. 만약 질의자가 정당한 사용자이고 복호화키(이메일의 경우: 개인키)를 가지고 있다면, SRS는 적어도 약한 인증은 만족해야 된다. [18].

Definition 4. *'$AT_W(AO)=1$'*일 확률이 무의미 함수[2]일 때, 안전한 검색 시스템 SRS (Secure Retrieval System)은 약한 인증을 보장한다.

5.2 복호화 주체가 복호화키를 가지고 있지 않은 경우

검색하고자 하는 민감한 정보가 자신이 생성한 혹은 자신을 위해 생성한 데이터가 아니고 다른 이의 데이터를 공유하는 경우가 모두 여기에 해당된다. *SSU* 모델과 *OSU* 모델에서 공개키 기반의 이메일 서버를 제외한 경우이다. 즉, 복호화 주체가 복호화키를 가지고 있지 않아 개체 인증 후 복호화를 위한 키가 사용자에게 부여되기 때문에, 이 경우 강력한 인증이 요구된다.

Definition 5. *'$AT_S(AO)=1$'*일 확률이 무의미 함수일 때, 안전한 검색 시스템 SRS (Secure Retrieval System)은 강한 인증을 보장한다.

VI. 내부 및 외부 공격자에 대한 프로토콜 시큐리티

암호화된 DB 검색 시스템의 전반적인 프로토콜의 기본적인 안전성에 대한 것으로서 외부 공격자(A_O) 뿐만 아니라 내부 공격자(A_I) 모두에 대해서도 안전해야 하는 보안에 관한 기본 요구 사항들에 관한 것으로서 정확성과 무결성이 있다 [23]. 내/외부 공격자 모두를 총칭하여 공격자 A라고 한다.

2) 무의미 함수 $\eta: N \rightarrow R$일때, 어떤 $c \in N$에 대해서, $\eta(n) < 1/n^c$ (모든 $n \geq n_c$ 에 대해서)을 만족하는 $n_c \in N$이 존재한다.

6.1 정확성(Correctness)

안전한 검색 시스템의 정확성은 그것을 구성하는 알고리즘들이 맞게 설계되었고, 그것을 운영하는 프로토콜이 정확하며, 네트워크를 통해서 전송 간 에러 없이 정확하게 동작하는 것을 말한다 [23]. 따라서 정확성은 질의자로 하여금 리턴된 결과들이 서버에 실제로 존재하고 질의에 매칭된 데이터임을 확신할 수 있게 한다.

Definition 6. 안전한 검색 시스템 프로토콜 *SRSP (SRS Protocol).*

안전한 검색 시스템 프로토콜 *SRSP (SRS Protocol)*은 셋업과 검색의 두 단계 *(SetSRS, RetrieveSRS)*로 이뤄지며, 이것은 SRS의 7가지 알고리즘에 대하여 질의/결과 스펙 (Query/Result spec)을 만족한다.

- $\boldsymbol{SRSP = (\ SetSRS,\ RetrieveSRS\ )}$
- $\boldsymbol{SetSRS = (\ SetUpSys(s),\ KeyGen(\lambda),\ SetUpEncDB(K,D)\ )}$
- $\boldsymbol{RetrieveSRS = (\ AuthUser(PI_U),\ SendQuery(PI_U,W,K),\ SearchDB(Q_w),\ Decrypt(R_c,K)\ )}$

Definition 7. 정확성 (Correctness).

안전한 검색 시스템 프로토콜 SRSP가 SRS의 7가지 알고리즘에 대하여 질의/결과 스펙 *(Q/R spec)*을 만족한다고 하자. 그러면, 시큐리티 파라메터 s와 시스템 파라메터 λ, 비밀키 집합 K에 의해서, 암호화된 데이터들의 집합 $E_K(D)$와 그것들의 인덱스 I_D는 다음에 의해서 서버 S에 저장되고;

$\boldsymbol{\{E_K(D),\ I_D\}_S \leftarrow SetSRS = SetUpEncDB(KeyGen(SetUpSys(s)))}$

SRSP는 다음을 만족해야한다;

- **정확한 인증 (correct authentication) :** $\boldsymbol{AuthUser(PI_U) = 1}$
- **정확한 검색 (correct retrieval) :** $\boldsymbol{R_c \leftarrow SearchDB(SendQuery(PI_U,W,k))}$
- **정확한 복호화 (correct decryption) :** $\boldsymbol{R_p \leftarrow RetrieveSRS = AuthUser(PI_U) \wedge Decrypt(SearchDB(SendQuery(PI_U,W,k)))}$

6.2 무결성(Integrity)

데이터 무결성 (Data Integrity)이란 모든 자료들이 허가된 방법에 의해서만 생성, 전송, 저장되고, 그 이후로 허가받지 않은 방법에 의해 변경되지 않는 것을 말한다. 따라서 데이터 무결성은 외부 공격자나 서버같은 내부 공격자가 데이터를 추가, 삭제, 변경하는 공격에 안전함을 보장한다 [18].

또, 서버가 악의적이거나 성실하지 못하여 질의를 전체 데이터 셑에 대해 수행하지 않고 결과의 일부만을 리턴할 수가 있다. 완전성(completeness)은 질의에 매칭하는 모든 결과를 리턴했다는 것을 질의자가 검증할 수 있음을 의미한다 [23].

이와 같은 두 성질을 참고로 하여, 안전한 검색 시스템에서의 무결성을 다음과 같이 정의한다.

Definition 8. 무결성 검증 메카니즘 (Iintegrity Checking Mechanism) Φ

안전한 검색 시스템 SRS에서 무결성 검증 메카니즘을 θ라고 하자. j번째 데이터 D_j에 대하여, D_j의 암호화된 데이터와 그것의 인덱스가 무결성 검증 테스트를 만족하면 '1'을 출력하고 그렇지 않으면 '0'을 출력하며 다음과 같이 표기한다.

- $\theta(I_j)=1 \text{or } 0$
- $\theta(E(D_j))=1 \text{or } 0$

서버 S에 총 n개의 데이터가 저장되어 있다고 하면, 안전한 검색 시스템 SRS의 무결성 검증 메카니즘은 다음과 같이 정의된다.

- **인덱스 검증 메카니즘** $\Phi_I=\prod_{j=1}^{n}\theta(I_j)$
- **암호화 데이터 검증 메커니즘**

$$\Phi_{E(D)}=\prod_{j=1}^{n}\theta(E(D_j))$$

Definition 9. 무결성 (Integrity).

안전한 검색 시스템 SRS에서, 서버에 저장된 총 데이터 개수 n이 N이라 하자. 악의적인 공격자 A는 저장된 데이테에 대하여 삽입/삭제/변경의 공격을 시도하고, 요청한 질의에 대하여 N개의 데이터에 대해서만 검색을 수행하려는 공격자라고 가정한다. 공격자 A에 의해 공격당한 서버 S의 무결성 검증 메카니즘이 다음 식을 만족할 확률이 무의미 함수 η이고 $n=N$이라면, 안전한 검색 시스템 SRS는 무결성을 보장한다고 말한다.

$$Adv_A=|\Pr[\Phi_{SRS}=\Phi_{E(D)}\times\Phi_I=1)\wedge(n=N)]|<\epsilon$$

- $\Phi_{SRS}=1\Leftrightarrow\Phi_{E(D)}=1\ \wedge\ \Phi_I=1$
- $\Phi_{E(D)}=1\Leftrightarrow\prod_{j=1}^{n}\theta(E(D_j))=\theta(E(D_1))\times\theta(E(D_2))\times\ldots\times\theta(E(D_n))=1$
- $\Phi_I=1\Leftrightarrow\prod_{j=1}^{n}\theta(I_j)=\theta(I_1)\times\theta(I_2)\times\ldots\times\theta(I_n)=1$

이 식은 공격자 A가 삽입/삭제/변경 등의 공격을 하였을 때, 무결성 검증 테스트를 통과할 확률이 무의미 합수, 즉 아무런 의미없는(negligible) 값 ϵ이라는 것이다. SRS가 무결성을 만족한다는 것, 즉 $\Phi_{SRS}=1$ 은 그것의 암호화 데이터 검증 메커니즘과 인덱스 검증 메카니즘 모두가 무결성을 만족한다는 것 ($\Phi_{E(D)}=1$와 $\Phi_I=1$)이고, $\Phi_{E(D)}=1$이고 $\Phi_I=1$인 것은 모든 데이터에 대한 검증 메카니즘이 무결성을 만족한다는 것이다.

VII. 내부 공격자에 대한 데이터 보호

데이터를 비밀키로 암호화한다는 것은 접근 제어뿐만 아니라 데이터 보호까지 동시에 가능하게 한다. 이 장에서 우리는 이런 외부 공격자에 대한 데이터 보호보다는 내부 공격자에 대한 데이터 보호에 집중한다. 내부 공격자에 안전한 시스템이라면, 외부 공격자에 대해서 안전한 것은 명백한 사실이며, 실제로 서버 매니저 같은 저장된 데이터에 접근 권한이 자유로이 허용된 내부 공격자에 의한 정보 유출은 심각한 사회적인 문제를 유발하기 때문이다.

암호화된 데이터를 저장하고 있는 악의적인 서버 매니저는 암호화되어 있는 인덱스 정보로부터 그 데이터가 어떠한 내용에 관한 것인지 알아내기 위해 시도하고, 질의에 대해 리턴되는 검색 결과들을 지속적으로 분석함으로써 그 데이터들의 내용을 캐내려 한다. 만약 이런 내부 공격자의 공격이 성공하여, 그 정보의 소유자 모르게 서버에 의해 정보가 외부로 노출되어 오용 및 악용된다면 이것은 심각한 프라이버시 침해가 되는 것이다. 이러한 의미에서 내부 공격자에 대한 데이터 보호를 프라이버시 보호로 해석하고, 안전한 검색 시스템에서 프라이버시 보호를 위한 방법으로 '인덱스 프라이버시(Index Privacy)'와 '질의 프라이버시(Query Privacy)' 이 두 가지로 나누어 생각해 본다. [24,25,27]에 기반하여, 우리는 시큐리티 게임

모델 SEM-CDA(semantic security against chosen data attack, 다른 말로, 'indistinguishability')을 정의한다.

Definition 10. 게임 SEM-CDA.

■ **셋업(Setup)**. 내부 공격자 A_I는 도전자에게 데이터 D를 준다. 그러면 도전자는 *KeyGen* 알고리즘을 동작시켜 비밀키를 생성하고 *SetUpEncDB* 알고리즘을 동작시켜 데이터 D 및 그것의 인덱스를 암호화한다. 이러한 과정은 A_I에 의해서 적응적(adaptive)으로 요청되어진다.

■ **질의(Queries)**. 공격자 A_I가 키워드 w를 질의하면 도전자 C는 *SendQuery* 알고리즘을 동작시켜 그것에 대한 암호화된 질의어 Q_w를 생성하여 공격자에게 준다. 이 Q_w를 가지고 공격자는 $w \in I$ 인지 아닌지 확인하기 위하여 인덱스 I 에 대해 *SearchDB* 알고리즘을 실행한다.

■ **도전(Challenge)**. 공격자 A_I는 질의 단계 후, 도전할 단어 $w_0(\in I_0 \subset D_0)$와 $w_1(\in I_1 \subset D_1)$을 선택한다. w_0와 w_1은 절대 이전에 질의 되어졌던 단어이면 안 된다. 공격자 A_I는 w_0와 w_1을 도전자 C에게 주고 C는 $b \in_R \{0, 1\}$를 선택하고 *SendQuery* 알고리즘을 실행시켜 w_b에 대한 암호화된 질의어 Q_{w_b}를 생성하여 다시 A에게 준다. 이후로 공격자 A_I는 w_0와 w_1에 대한 질의를 하여서는 안 된다.

■ **반응(Response)**. 마지막으로 공격자 A_I는 b를 추측하여 b' 을 출력한다. 이 게임에서 공격자 A_I가 이길 확률 Adv_{A_I}을 다음과 같이 정의한다; $Adv_{A_I} = |\Pr[b = b'] - \frac{1}{2}|$, 여기서 확률은 공격자와 도전자의 동전 던지기(coin toss)이다. 만약 $Adv_{A_I} > \epsilon$ 이라면, 공격자 A_I는 $\epsilon -$ 승률$(advantage)$ 을 가졌다고 말한다.

우리는 공격자 A_I가 이길 확률 Adv_{A_I}가 무의미 함수라면, '안전한 검색 시스템 SRS는 게임 SEM-CDA에 대하여 안전하다'라고 말한다.

Definition 11. 안전한 검색 시스템 SRS의 인덱스가 게임 SEM-CDA에 대해서 안전하다면 SRS는 인덱스 프라이버시를 보장한다고 말한다.

인덱스 프라이버시라는 것은 암호화되어 저장되어 있는 인덱스를 내부 공격자가 관찰 및 분석을 통해 그와 연관된 데이터의 내용을 알아내고자 하는 공격에 안전함을 의미한다. 결국 인덱스 프라이버시를 제공하는 시스템에서 공격자는 어떤 두 문서에 관한 인덱스가 공통된 키워드를 가졌다 할지라도 그 사실을 알지 못한다.

Definition 12. 안전한 검색 시스템 SRS의 질의어 Q_w가 게임 SEM-CDA에 대해서 안전하다면 SRS는 질의 프라이버시를 보장한다고 말한다.

질의 프라이버시라는 것은 서버 매니저가 요청받은 질의에 대해 검색을 수행하여 그것의 결과를 버리지 않고 저장해 두고, 이런 누적된 결과들을 분석함으로써 저장된 데이터의 내용을 알아내려고 시도하는 공격에 안전함을 의미한다. 이 때 공격자는 만약 질의어가 같은 사용자가 같은 키워드를 반복해서 질의하였다 하더라도 그 사실을 알지 못하게 된다.

Definition 13. 완벽한 검색 프라이버시(Perfect Retrieval Privacy).

안전한 검색 시스템 SRS가 접근제어(약한/강한 인증), 프로토콜 안전성(무결성, 정확성), 데이터 보호(인덱스 프라이버시, 질의 프라이버시), 이 모두를 만족하면, 완벽한 검색 프라이버시를 보장한다고 말한다.

VIII. 안전성 분석 적용

이 장에서는 앞서 정의한 보안/프라이버시 모델을 이용하여 기존에 제안되어진 스킴들의 안전성을 분석해 본다. Agrawal et al.의 'Order Preserving Encryption for Numeric Data' [2]이 그 대상 논문으로서, Agrawal et al.의 스킴은 이미 [1]에서 안전하지 않다고 증명되었지만 그것의 타당성 여부를 좀 더 세부적으로 분석해 보고자 한다.

8.1 OPES (Order Preserving Encryption for Numeric Data)

이것의 응용 환경의 특징은 공격자는 그 데이터베이스에 저장된 수량 데이터(numeric data)의 분포와 같은 사전 지식을 알고 있으면 안 되고, 그가 임의적으로 생성한 값으로 암호화나 복호화를 할 수 없으며, 서버가 비신뢰적인 경우로 서버단에서 복호화 과정이 없다.

8.1.1 외부 공격자에 대한 접근제어

OPES는 그들의 논문에서 DB2에 적용한 실험 측

정치를 보여준다. 이처럼 OPES는 현재의 상용 DB에도 적용 가능하며, 안전한 인증 시스템을 보장하는 어느 DB에서나 적용 가능하기 때문에 외부 공격자에 대한 접근 제어가 가능하다고 할 수 있다.

8.1.2 내/외부 공격자에 대한 프로토콜 안전성

1. 정확성 (Correctness)

OPES도 역시 접근제어가 가능하기 때문에 정확한 인증 (correct authentication)을 만족한다고 할 수 있으며, 그들의 논문 섹션 8. EVALUATION에서 실험을 통하여 다음을 만족함을 보여준다.

$\{\{EK(D), ID\}_S \leftarrow SetSRS = SetUpEncDB(KeyGen(SetUpSys(s)))$

- 정확한 검색 (correct retrieval) : $R_c \leftarrow SearchDB(SendQuery(PI_U, W, K))$
- 정확한 복호화 (correct decryption) : $Rp \leftarrow RetrieveSRS = AuthUser(PI_U) \wedge Decrypt(SearchDB(SendQuery(PI_U, W, K)))$

2. 무결성 (Integrity)

OPES는 무결성 체크 알고리즘을 따로 찾아볼 수 없다. 주어진 프로토콜 그대로 무결성을 분석해 보면, 데이터의 추가시 완전성(completeness)은 보장될 수 있다. 하지만, 데이터의 삭제나 변경 시에는 검색의 완전성 (completeness)을 만족시킬 수 없으므로 무결성을 보장할 수 있다고 말할 수는 없다.

8.1.3 내부 공격자에 대한 데이터 보호

1. 인덱스 프라이버시 (Index Privacy)

이 스킴의 특징은 암호화 후에도 원래 데이터의 대소 관계가 유지된다는 것이다. 따라서 공격자는 인덱스를 보고 그 데이터들의 정확한 원래의 값으로 복호화하는 것은 불가능 하지만 대소 관계는 알 수 있다. 따라서 이 논문은 인덱스 프라이버시를 만족하지 못한다.

2. 질의 프라이버시 (Query Privacy)

이 논문에서는 질의 알고리즘에 관한 특별한 언급이 없다. 이것은 일반적인 DBMS의 질의 명령어로 검색 가능하다는 것이며, 일반적인 질의 명령어는 암호화된 DB가 아닌 평문에 관한 것이기 때문에, 질의 프라이버시에 관해서 고려하지는 않았다. 좀 더 구체적으로 말하자면, 질의 프라이버시를 보장하기 위해서는 질의어에 매번 다른 난수적 요소가 첨가되어야 하는데, OPES는 그러한 알고리즘을 사용하지 않았고, 인덱스와 문서 암호화에 사용되는 함수가 결정적 함수 (deterministic function)이기 때문에 같은 질의어를 같은 키 값으로 암호화한 값은 항상 같게 되어 질의 프라이버시를 보장할 수 없다.

8.1.4 OPES 분석 결과

OPES는 데이터보호와 무결성은 보장할 수 없지만 최소 보안요구 조건인 접근제어와 정확성은 보장한다. 실제로 무결성 알고리즘을 가지고 있는 스킴은 많지 않고 데이터보호를 보장하는 알고리즘 역시 현실적으로 소수에 지나지 않는다. 결과적으로 [1]에서 OPES에 대한 분석은 그 스킴 자체와 현실적인 환경에 대한 세심한 분석 없이 한 결과라고 할 수 있으며, 오히려 [1]의 서버가 신뢰적인 개체라는 가정 하에 서버단에서의 복호화 과정이 있는 것이 더 위험하다. 그리고 응용 환경의 상황에 따라 적합한 가정은 다를 수 있기 때문에 이런 사실만으로 어느 스킴이 더 안전하고 덜 안전하다고 말할 수는 없다.

IX. 제언

모든 암호화 모듈을 분석할 때는 그것의 이론적인 안전성 뿐만 아니라 실제 환경에서 적용시 그 효용성에 대해서 같이 생각해 봐야 한다. 현재 시스템과의 연동 가능성 및 효율성 등을 말이다. 너무 효율성만을 고려하여 안전하지 않은 모듈을 사용한다면, 그것은 사용하지 않음만 못하다. 따라서 모든 기업이나 조직들은 적절한 안전성하에서 실제 상황에 적합한 효율성도 역시 만족하는 모듈을 원한다. 따라서 모든 스킴에 대한 정확한 분석이 필요하며 이 정확한 분석을 위해서는 어떠한 기준이 필요하고 그 스킴의 응용 환경에 대해서도 심오하게 고려하여야 한다. 하지만, 세상에 존재하는 모든 응용 환경을 포괄할 수 있는 하나의 일반적인 기준을 제시한다는 것은 실로 불가능한 일이다. 따라서 우리는 모든 암호화된 검색 시스템을 포괄할 수 있는 공통적인 안전성 및 프라이버시에 대한 요구 사항을 모델링 하였다. 우리가 정의한 13가지 정의들을 모두 만족해야 된다는 의미가 아니다. 조직의 목적이나 환경의 특색에 따라 적정한 레벨을 이 중에서

선택할 수 있다는 것이다. 하지만 여기서의 문제점은 어떠한 환경에서 어떠한 항목들을 만족해야 하는가이다. 너무나 다양한 응용 환경과 각 응용 환경에 따라 조직의 목적이나 상황이 다를 수 있기 때문에 하나하나 나열한다는 것은 불가능하므로 최소한으로 만족해야 할 사항을 제언하고자 한다.

현재 상용 모듈의 대부분은 약한 인증과 정확성은 모두 만족한다. 오라클과 Microsoft SQL Server 2008 등은 강한 인증과 안전한 키 관리 시스템까지 지원한다. 하지만 내부공격자에 대한 안전성까지 만족하는 상용화 모듈은 아직까지는 없다. 우리는 안전한 검색 시스템이 만족해야 할 최저 보안 요구 조건으로 약한 인증과 정확성을 제언한다. 요즘 상용화 제품이 강한 인증과 키 관리 시스템을 지원하고 있긴 하여도 응용 환경에 따라 이것이 필요한 것일 수도 있고 아닐 수도 있기 때문에 우리는 이것을 선택 사항으로 두겠다. 그리고 보안 요구 사항으로 오직 접근제어와 정확성만을 생각한다면 안전성 보장에 있어 다소 허술하다고 생각될 수도 있으나 우리가 정의한 정확성은 상당히 다양한 보안적 요소를 포함한다. 암호화 알고리즘의 안전성, 즉 암호화 데이터베이스의 기밀성과 데이터 전송간의 안전성 등등 안전한 검색을 위한 모든 하드웨어적/소프트웨어적 문제를 포괄한다. 이런 최소한의 기본적인 안전성 위에 시스템 환경과 조직의 특수한 목적에 따라 추가적인 안전성을 고려해야 하는 것이다. 이러한 고찰 없이 오직 자신들의 환경에 맞게 제시된 보안 요구 사항으로 다른 스킴을 평가한다면 OPES와 같은 오류를 낳게 되는 것이다.

X. 결론

우리는 Zdonik et al.의 OPES에 대한 안전성 증명을 검토하고 몇 가지 문제점을 재분석하였다. 이러한 오류를 다시 반복하지 않고 어떤 시스템을 정확하게 분석하기 위해서 그 시스템과 응용 환경에 대해 합당한 기준을 제공하기 위해, 우리 연구는 일차적으로 모든 암호화 검색 시스템에 적용할 수 있는 시큐리티와 프라이버시에 대한 포괄적인 기준을 정형화하여 거기서 지켜야 할 최소한의 보안 요구 사항을 제공하고, 그 환경의 특성 및 조직의 정책에 따라 추가적인 안전성을 고려할 것을 제언한다. 우리의 연구는 이러한 선택을 원하는 사용자와 안전한 검색 시스템을 디자인하는 설계자에게 일종의 지침을 제공할 수 있다.

향후 연구 과제로서 모든 응용 환경에 적용 가능한 등급별 보안 및 프라이버시 모델을 설계할 필요가 있을 것으로 보여진다. 이 논문에서 우리는 공격자 유형에 의한 안전성을 분류하여 모델을 제시하였지만, 차후의 연구에서는 안전성 수준을 기본적인 최하부터 최상까지 보다 세부적으로 등급을 나누고, 응용 환경들의 보안 요구 사항 역시 등급을 주어 그것의 안전성 수준을 맵핑한 목록을 만들어 많은 사람들이 지침서로 활용하도록 하는 것을 목표로 한다.

참고문헌

〔1〕 T. Ge and S. Zdonik, "Fast, secure encryption for indexing in a column-oriented DBMS", Proceedings of the 23rd ICDE, pp. 676-685, Apr. 2007.

〔2〕 R. Agrawal, J. Kiernan, R. Srikant and Y. Xu. "Order preserving en-cryption for numeric data", Proceedings of the ACM SIGMOD, pp. 563-574, June 2004.

〔3〕 R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Hippocratic databases", In The 28th International Conference on Very Large Databases (VLDB), pp. 143-154, Aug. 2002

〔4〕 L. Ballad, S. Kamara, and F. Monrose, "Achieving efficient conjunctive keyword searches over encrypted data", Proceedings of the ICICS 2005, LNCS 3783, pp.414-426, Dec. 2005.

〔5〕 E. Mykletun and G. Tsudik, "Aggregation Queries in the Database-As-a-Service Model", Proceedings of the DBSEC 2006, pp.89-103, Aug. 2006.

〔6〕 G. Ozsoyoglu, D. Singer, and S. Chung, "Anti-tamper databases: Querying encrypted databases", Proceedings of the IFIP Conference 2003 on Database Security, pp. pp.133-146, Aug. 2003.

〔7〕 S. Chung and G. Ozsoyoglu, "Processing aggregation queries over encrypted databases", Proceedings of the ICDE 2006, pp. 98, Apr. 2006.

〔8〕 H. Hacigumus, B.Iyer, and S. Mehrotra, "Efficient execution of aggregation queries

over encrypted relational databases", Proceedings of the DASFAA, LNCS 2793, pp.125-136, Mar. 2004

〔9〕 H. Hacigumus, B.R. Iyer, L. Chen, and S. Mehrotra, "Executing SQL over encrypted data in the database-service-provider model", Proceedings of the ACM SIGMOD, pp. 216-227, June 2002.

〔10〕 L. Ballard, M. Green, B. de Medeiros, F. Monrose, "Correlation-resistant storage via keyword - searchable encryption", SPAR Technical Report, TR-SP-BG-MM-050705.

〔11〕 J. Byun, H. Rhee, H. Park, and D. Lee, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data", Proceedings of the SDM2006, LNCS 4165, pp. 75-83, Sep. 2006.

〔12〕 K. Bennett, C. Grothoff, T. Horozov, and I. Patrascu, "Efficient sharing of encrypted data", Proceedings of the ACISP02, pp 107-120, July 2002.

〔13〕 D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data", Proceedings of the TCC 07, LNCS 4392, pp. 535-554, Feb. 2007.

〔14〕 D. Boneh, G. Crescenzo, R. Ostrovsky, and G. Persiano, "Public-key encryption with keyword search", Proceedings of the Eurocrypt04, LNCS 3027, pp. 506-522, May 2004.

〔15〕 Y. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data", IACR ePrint 2004-051, 2004.

〔16〕 N.Cao, C.Wang, M.Li, K.Ren, and W.Lou, "Privacy-preserving multikeyword ranked search over encrypted cloud data". Proceedings of the INFOCOM, pp. 829 - 837, April 2011.

〔17〕 J. Gemmell, G. Bell and R. Lueder, "MyLifeBits: a personal database for everything", Communications of the ACM, vol. 49, Issue 1, pp. 88-95, Jan. 2006.

〔18〕 A,J.Menezes, P.C. Oorschot and S.A. Vanstone, Handbook of applied cryptography, CRC Press., Oct. 1996.

〔19〕 M. Klemettinen, Enabling technologies for mobile services : The MobiLife Book, Wiley, Nov. 2007.

〔20〕 J. Camenisch and A. Lysyanskaya, "A formal treatment of onion routing". Proceedings of the Crypto 2005, LNCS 3621, pp. 169 - .187, Aug. 2005.

〔21〕 W. Chen., Z. Liu, and A. Riabov, "A soft constraint privacy model based on identifiability," Proceedings of the 31st Annual International Computer Software and Applications Conference, pp. 675-980, July 2007.

〔22〕 D.E. Denning, "Database security", Annual Review of Computer Science, vol 3, pp. 1-22, 1988

〔23〕 M. Gertz and S. Jajodia, Handbook of database security applications and trends, Springer; 1st Ed., 2007.

〔24〕 E. Goh, "Secure indexes", IACR ePrint 2003-216, 2003.

〔25〕 P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data", Proceedings of the ACNS04, LNCS 3089, pp. 31-45, June 2004.

〔26〕 H. Park, J. Byun, and D. Lee, "Secure index search for groups", Proceedings of the TrustBus 05, LNCS 3592, pp. 128-140, Aug. 2005.

〔27〕 H. Park, B. Kim, D. Lee, Y. Chung, and J. Zhan, "Secure similarity search", Proceedings of the IEEE Grc 2007, pp. 598-604, Nov. 2007.

〔28〕 H. Park, J. Park, and D. Lee, "PKIS: Practical keyword index search on cloud datacenter", EURASIP Journal on Wireless Communications and Networking, vol. 84, no. 8, pp. 1364-1372, Aug. 2011.

〔29〕 H. Rhee, S. Kim and D. Lee, "A survey

of keyword search on encrypted data", Proceedings of the IJWISA 2008, pp. 49 - 58, Feb. 2008.

〔30〕 D.X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data", Proceedings of the 2000 IEEE Symposium on Security and Privacy, pp. 44-55, May 2000

〔31〕 S. Vaudenay, "On privacy models for RFID", Proceedings of the ASIACRYPT 2007, LNCS 4833, pp. 68 - .87, Dec. 2007

〔32〕 C.Wang, N.Cao, J.Li, K.Ren, and W.Lou, "Secure ranked keyword search over encrypted cloud data", Proceedings of the IEEE 30th International Conference on Distributed Computing Systems, pp. 253-262, June 2010

〔33〕 R. Waisenberg, E. Shmueli, and E. Gudes, "A Structure preserving database encryption scheme", Proceedings of the Secure Data Management(SDM), LNCS 3178, pp. 28-40, Sep. 2004.

〔34〕 P. Wang, H. Wang, and J. Pieprzyk, "Keyword field-free conjunctive keyword searches on encrypted data and extension for dynamic groups", Proceedings of the CANS 2008, LNCS 5339, pp. 178-195, Dec. 2008

〔35〕 S. Zerr, D. Olmedilla, W. Nejdl and W. Siberski, "Zerber+R: Top-k retrieval from a confidential index", Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, pp. 439-449, Mar. 2009

〔36〕 X. Zhu and J. W. Atwood, "A web database security model using the host identity protocol", 11th International Database Engineering and Applications Symposium, In the proceedings of IDEAS 2007, pp. 278-284, Sep. 2007.

〔37〕 Computer Security Institute, CSI/FBI computer crime and security survey, annual series.

〈著者紹介〉

박 현 아 (Hyun-A Park) 학생회원
2003년 2월: 고려대학교 수학과 졸업
2005년 2월: 고려대학교 정보보호대학원 석사
2010년 2월: 고려대학교 정보보호대학원 박사
2011년 11월~현재: 한국과학기술정보연구원(KISTI) 연구원
〈관심분야〉 DB 보안, PET(Privacy Enhancing Technology), 암호 프로토콜, 클라우드 보안

이 동 훈 (Dong Hoon Lee) 정회원
1983년 8월: 고려대학교 경제학사
1987년 12월: Oklahoma University 전산학 석사
1992년 5월: Oklahoma University 전산학 박사
1993년 3월~1997년 2월: 고려대학교 전산학과 조교수
1997년 3월~2001년 2월: 고려대학교 전산학과 부교수
2001년 2월~2006년 8월: 고려대학교 정보보호대학원 부교수
2006년 9월~현재: 고려대학교 정보보호대학원 교수, 부원장
〈관심분야〉 암호프로토콜, 암호이론, USN 이론, 키이론, 임베디드 보안

정 택 영 (Taik Yeong Chung) 정회원
1984년 2월: 계명대학교 전산학과 졸업
1986년 2월: 한국과학기술원 전산학 석사
2010년 2월: 광운대학교 경영정보학 박사
1986년 1월~현재: 한국과학기술정보연구원(KISTI) 연구원
〈관심분야〉 정보화전략 수립 및 계획, EA(Enterprise Architecture), S/W개발 프로젝트 관리, 정보보안 및 개인정보보호