한국시뮬레이션학회 논문지
Vol. 24, No. 4, pp. 153-161 (2015. 12)
http://dx.doi.org/10.9709/JKSS.2015.24.4.153
ISSN 1225-5904

완전동형암호기반 프라이버시 보호 Top-k 위치정보서비스

허미영 · 이윤호*

Privacy Preserving Top-k Location-Based Service with Fully Homomorphic Encryption

Miyoung Hur · Younho Lee*

ABSTRACT

We propose a privacy-preserving location-based service (LBS) which supports top-k search service. The previous schemes hurt the privacy of either the user and the location of the objects because they are sent to the LBS server in a plaintext form. In the proposed method, by encrypting them with the fully-homomorphic encryption, we achieved the top-k search is possible while the information on them is not given to the LBS server. We performed a simulation on the proposed scheme with 16 locations where k is 3. The required time is 270 hours in a conventional desktop machine, which seems infeasible to be used in practice. However, as the progress of the hardware, the performance will be improved.

Key words : Privacy protection, Top-k search, Location based service, Security

요 약

Top-k 위치정보서비스는 사용자의 위치로부터 가장 가까운 k개의 장소를 반환하는 서비스이다. 기존의 방법들은 사용자의 위치 정보가 LBS Server에 그대로 노출되어 사용자의 프라이버시 훼손의 문제가 있다. 본 논문에서는 완전동형암호를 사용하여 Top-k 위치정보서비스 사용자의 프라이버시를 보호하는 방안을 연구한다. 제안 방법에서는 사용자의 위치 정보가 포함된 질의와 Database의 위치 데이터 정보를 암호화한다. LBS Server는 완전동형암호를 이용해 암호화된 질의 위치 정보와 암호화된 위치 데이터로 거리 계산을 수행한다. LBS Server는 계산 결과를 암호문 상태에서 비교하여 사용자의 위치로부터 가장 가까운 위치가 저장되어 있는 k개의 암호문을 결과로서 도출한다. 결과는 LBS Server로부터 사용자에게 반환되며, 사용자는 이를 복호화하여 자신의 질의 결과를 확인한다. 본 방법에서는 Database의 위치 데이터와 사용자의 질의 정보가 모두 암호화된 상태로 Top-k 위치정보서비스를 제공하므로 LBS Server에 대해 사용자와 위치 데이터 정보의 프라이버시가 보존된다. 시뮬레이션에서는, 16개의 위치 정보에 대하여 질의와 거리 연산을 수행하여 사용자의 질의로부터 가장 가까운 3개의 위치를 알아내는 과정을 수행하였다. 그 결과 일반적인 데스크탑 환경에서 약 270시간이 걸려 단기간 내의 실용화는 어려울 것으로 예상되나 이러한 성능 문제는 하드웨어의 발전과 함께 개선될 것이라 생각된다.

주요어 : 프라이버시 보호, Top-k 검색, 위치정보서비스, 보안

1. 서 론

최근 모바일 사용자가 늘어나고, 그에 따른 위성항법장치(GPS) 사용이 늘어나면서 Location-based Service (LBS)가 각광을 받고 있다. 사용자의 위치 정보를 기반으로 지도를 활용한 내비게이션 서비스, 근거리 장소 찾기 서비스 등 LBS는 여러 사례로 활용되고 있다.

검색 기술을 LBS에 적용한 예시로 Semantics 분류 기술[1], Time Ontology, Content Ontology[2] 등 여러 방식

* 이 논문은 2015년도 정부의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2013R1A1A2011754).

Received: 14 December 2015, **Revised:** 20 December 2015, **Accepted:** 20 December 2015

***Corresponding Author:** Younho Lee
E-mail: younholee@seoultech.ac.kr
Department of Industrial and Systems Engineering, SeoulTech

이 있지만 본 연구에서는 그 중 하나인 Top-k 방식을 사용한다[3]. Top-k 방식이란, LBS Server가 사용자의 위치 정보와 가장 가까운 k개의 위치를 사용자에게 반환하는 방식이다. 이러한 Top-k 위치 검색은 사용자가 원하는 k개의 정보를 반환받을 수 있어 편리하고 유용한 기술이다. k는 가변적이지만, 원활한 서비스를 사용하기 위해 k는 10 이하의 작은 수로 한다.

Top-k 방식의 위치정보서비스는 보통 사용자, LBS Server, Database의 구성 요소로 이루어져 있다. Database는 LBS Server에 위치 데이터를 제공하고, 사용자는 자신의 위치 정보와 관심 있는 객체의 수 k를 LBS Server에게 질의한다. LBS Server는 Database의 위치 데이터와 질의에 포함된 사용자의 위치 정보를 바탕으로 거리 계산을 수행하여 사용자의 위치와 가까운 k개의 객체를 찾아낸다. 모든 위치 데이터의 계산이 완료된 후 LBS Server는 선택된 k개의 객체의 유형과 위치를 사용자에게 결과로서 반환한다.

하지만 이러한 Top-k 방식을 이용하기 위해 LBS Server에 전송되는 사용자의 질의는 자칫 프라이버시 노출로 직결될 수 있다. 예를 들어 LBS Server는 사용자의 위치정보와 객체 정보를 연결시켜 중대한 프라이버시 위협을 만들어낼 수 있다. 사용자는 LBS Server로부터 프라이버시 보호를 보호하면서 정확한 Top-k 서비스를 제공받기 위해 LBS Server에 자신의 위치 정보 노출을 원하지 않을 수 있다. 또한 Database도 위치 데이터를 보호하며 LBS Server에 위치 정보를 전송한다면, LBS Server는 Database의 위치 정보와 사용자의 거리 계산 결과를 확인할 수 없어 사용자의 질의 위치를 추측할 수 없을 것이다. 즉, LBS Server는 어떠한 위치 정보도 알지 못하게 만들어 이상적으로 높은 보안 수준을 만족하는 Top-k 서비스를 제공할 수 있을 것이다.

이러한 LBS Server로부터의 프라이버시 보호와 관련된 연구는 다음과 같다. 첫 번째로 LBS Server가 Database와 Top-k의 결과의 임의 조작을 방지한 논문이 있다[4, 5]. 해당 논문에서는 사용자에 의한 결과 검증으로 악의적인 LBS Server의 결과 조작은 해결했지만, Database의 위치 데이터와 사용자의 위치 정보가 LBS Server에 노출되는 문제가 남아 있다. 이를 보완하여 LBS Server에게 Database의 위치 데이터만 그대로 전송하고 사용자의 위치 정보는 숨기는 또 다른 연구도 있다[6]. 사용자는 자신의 위치 정보를 LBS Server에 질의 시, 동료 사용자의 위치 정보를 이용하여 자신의 위치를 모호하게 한다. 하지만 만약 충분한 수의 동료 사용자가 없다면 유일한 사용자의 위치 정보는 그대로 LBS Server로 노출되는 문제점이 있다. 그와 유사한 연구로 동료 사용자의 정보를 사용하여 사용자의 위치를 LBS Server의 공개키로 암호화한 후, 사용자의 위치를 LBS Server에 전송하는 연구도 있다[7]. 질의 과정에서 사용자의 위치가 암호화되었기 때문에 위치 데이터의 전송 보안성이 높아졌지만, LBS Server가 개인키로 사용자로부터 전송된 암호문을 복호화할 수 있어 결국 사용자의 위치 정보가 노출되는 문제가 있다. 동료 사용자를 사용하지 않고 LBS Server에 사용자의 위치 대신 사용자의 영역 정보를 전송해 사용자의 프라이버시를 보호하려는 논문도 있다[8, 9]. 이 논문에서는 사용자의 위치 정보를 은닉하여 LBS Server가 사용자의 위치를 정확히 알 수 없도록 만들었다. 하지만 만약 연속적인 사용자의 질의가 발생한다면, LBS Server는 사용자의 위치를 유추할 수 있다. 이는 사용자의 프라이버시를 보호했다고 판단되기 어려울 뿐더러 LBS Server가 Database의 위치 데이터를 알고 있는 문제도 남아있다. 또한 Top-k 방식을 사용하기 위해 사용자, LBS Server, Database 외에 익명 서버를 추가해 사용하는 방식도 연구되었다[10]. 해당 연구는 익명 서버를 이용하여 LBS Server에 사용자의 위치 정보를 노출하지 않으려고 시도했다. 하지만 이론상 익명 서버가 존재할 수는 있지만, 현실적으로 신뢰할만한 익명 서버를 만드는 것은 매우 어렵다. 사용자가 LBS Server에 위치 정보를 포함한 질의를 전송하지 않고, 가장 가까운 위치 서비스를 받도록 Database 암호 검색 기술인 private information retrieval(PIR)을 LBS에 적용시킨 또 다른 사례가 존재한다[11, 12]. 본 방법에서 LBS Server는 단일 결과만을 제공하기 때문에 사용자가 자신의 위치로부터 가까운 k개의 장소를 알고자 한다면, 사용자는 검색을 k번 반복해야하는 단점이 있다.

결론적으로 LBS Server로부터 Database의 위치 데이터와 사용자의 위치 정보 노출 문제를 동시에 해결하는 방안은 아직 제시되지 않았다.

이에 본 논문에서는 Database 위치 정보와 사용자의 위치 프라이버시를 보호하며 Top-k 위치정보서비스를 이용하는 방법을 제안한다. 본 방법에서는 Database의 위치 데이터와 사용자의 질의를 완전동형암호로 암호화한다. 일반 암호와는 달리 데이터가 완전동형암호로 암호화되어 있기 때문에, LBS Server는 어떠한 데이터도 알지 못한 채 사용자의 위치로부터 가까운 k개의 객체의 위치를 계산할 수 있다. 계산 결과는 암호화되어 사용자에게 전송되며, LBS Server로부터 결과를 수신한 해당 사용자만 암호문을 복호화하여 결과를 확인할 수 있다.

제안 방법에 대한 간략한 설명은 다음과 같다. 기존 Top-k 위치정보서비스 구성요소인 사용자, Database, LBS Server외에 외부의 키 서버가 구성요소로 존재한다. 외부의 키 서버는 사용자, Database, LBS Server에 동일한 공개키를 제공하고 그에 해당하는 개인키는 사용자에게만 제공한다. 사용자에게 제공된 개인키는 LBS Server가 제공하는 서비스 과정 동안에만 사용 가능하다. 사용 기간이 만료된 사용자의 개인키는 회수되기 때문에 모든 위치정보와 데이터는 외부로 노출되지 않는다. Database는 위치 데이터를 공개키로 암호화한 후 LBS Server에게 전송하고 사용자는 Top-k 위치정보서비스를 이용하고자 할 경우, 자신의 위치 정보를 포함한 질의를 공개키로 암호화한 후 LBS Server에 질의한다. 이 때 Database의 위치 데이터와 사용자의 질의가 동일한 공개키로 암호화되어 있기 때문에 LBS Server는 모든 데이터를 모른 채로 완전동형암호를 이용하여 연산을 수행할 수 있다. LBS Server는 완전동형암호를 사용하여 Database의 위치 데이터와 질의에 포함된 사용자의 위치를 이용해 암호문 상태에서 거리를 계산한다. 계산 결과는 암호화된 상태에서 비교가 가능하며 가장 작은 거리 값에 대응되는 k개의 위치가 저장된 암호문이 사용자에게 결과로서 반환된다. 사용자는 LBS Server로부터 받은 암호문을 사전에 외부의 키 서버로부터 획득한 개인키로 복호화하여 결과를 확인할 수 있다.

본 제안 방법에서는 LBS Server가 Database의 위치 데이터와 사용자의 질의 정보를 암호화된 상태로 처리하므로 해당 정보들을 모두 알 수 없는 장점이 존재한다.

제안 방법의 유효성을 증명하기 위해 본 논문에서 완전동형암호를 이용한 테스트 환경을 구축했다. 테스트 환경은 Ubuntu, Intel Xeon processor Quad-core 1600 MHz, 64GB RAM, HDD 916.8GB 환경에서 진행되었다. 완전동형암호 연산을 위해 gmp-6.0.0, ntl-9.0.1 기반의 HElib library를 사용했고, C++언어를 사용했다. Database의 위치 데이터와 사용자의 질의는 2차원의 정수로 표현했으며 Database의 위치 데이터 크기는 16, 사용자가 요구하는 k는 3으로 설정하였다. 실험에서 사용된 Database의 위치 데이터는 {(25, 67), (46, 44), (21, 35), (21, 66), (32, 75), (36, 86), (11, 11), (19, 10), (53, 11), (47, 64), (38, 27), (18, 26), (28, 47), (22, 22), (27, 11), (22, 13)}로 하였고, 질의에 포함된 사용자의 위치 정보는 (6, 7)로 설정하였다. 실험을 위해 사전에 정의한 파라미터 L=17, B=28는 보안 파라미터로 설정하였고, 이는 공개키 쌍을 생성하는데 사용되었다. 실험에서 계산된 거리는 암호화된 상태에서 두 개씩 비교되며 가장 작은 수의 데이터부터 저장되도록 만들었다. 실험 결과 계산된 수 중 가장 작은 정수 41, 178, 272가 저장된 위치 데이터가 결과 k로 나왔고, 크기 비교의 시간은 약 11일 5시간 18분이 소요되었다.

위의 실험 결과로부터 본 방법의 구동을 위한 시간이 많이 소요되는 문제를 발견하였다. 그러나 이 문제는 향후 하드웨어 성능이 발전되면 개선될 것으로 예상된다. 프라이버시 노출로 인한 피해 비용은 나날이 증가하는 추세이기 때문에 본 방법은 프라이버시 보호의 해결책으로써 의의가 있다고 사료된다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문의 문제 해결 방안을 제시한다. 3장에서는 제안된 방안을 바탕으로 시뮬레이션을 수행하여 결과를 기술하고 4장에서는 결론을 서술하고자 한다.

2. 제안 방법 상세

2.1 정의 환경

본 논문에서 사용자, LBS Server, Database가 만들고 사용하는 암호문을 이하 Cipher라 정의한다. LBS Server에 대해 위치 프라이버시를 보호하며 Top-k 위치정보서비스를 제공받기 위해, 사용자와 Database는 먼저 자신의 위치 데이터를 암호화시켜야 한다. 위치 데이터가 저장된 암호문을 만들기 위해 사용자와 Database는 사전에 공유된 공개키가 필요하다. 본 논문에서는 공개키 쌍을 만들기 위해 외부의 키 서버를 사용하고, 외부의 키 서버는 사용자와 LBS Server, Database의 요청(call)이 없어도 각 주체에 미리 키를 배포한다고 가정한다.

Top-k 위치정보서비스를 이용하기 전 사용자와 LBS Server, Database는 사전에 외부의 키 서버로부터 동일한 공개키를 제공받는다. 이하 공개키를 PK라고 지칭한다. PK는 위치 정보 데이터를 암호화하고, LBS Server의 결과를 저장하는 암호문 Cipher를 생성하며 암호화된 데이터를 전달하는데 사용된다.

반면, 외부의 키 서버는 개인키를 오직 사용자에게만 제공한다. 이하 개인키를 SK라고 지칭한다. 사용자는 LBS Server로 전송받은 암호화된 결과를 확인하는데 SK를 사용한다. SK의 유효기간은 Top-k 위치정보서비스가 진행되는 한 번의 프로세스에서만 유효하며, 만료된 키는 추후 이용하는 Top-k 위치정보서비스에서 다시 사용할 수 없다.

Top-k 위치정보서비스의 목적은 1) 데이터를 암호화함으로써 데이터가 노출되지 않도록 하는 것 2) 암호화된

상태로 연산을 가능하게 하는 완전동형암호를 사용함으로써 사용자가 위치 프라이버시를 보호하며 Top-k 위치정보서비스를 사용 가능하게 만드는 것이다.

LBS Server로 전달되는 Database의 위치 데이터는 지도 정보로써 주요 객체의 위치가 암호화되어 제공된다. 이 때, 사용자의 위치 정보로부터 해당 객체들 간의 거리 계산이 발생한다. 편의상 Database의 위치 데이터 집합을 D라고 하고, D는 위치 데이터 를 d개 가진다고 한다. 즉, ={1, 2, ... , d}를 갖는 범위이고 이를 이용하여 D= $\{p_1, p_2, p_3, ..., p_d\}$로 표기한다. LBS Server에 위치 데이터를 노출시키지 않도록 하기 위하여 D의 각 좌표는 암호화되어 LBS Server에 전송된다. LBS Server는 Top-k 연산을 수행할 때 정확한 결과 값을 반환하기 위하여 암호화 상태에서 사용자의 질의 위치와 D를 모두 거리 계산해야한다. 따라서 D의 범위가 클수록 계산해야하는 데이터가 늘어나게 되므로 연산시간은 길어질 수밖에 없다.

LBS Server에 전달되는 사용자의 질의는 사용자의 위치 데이터 (x_u,y_u)와 Top-k의 결과를 도출하기 위한 파라미터 정수 k를 포함한다. 편의상 사용자의 질의를 Q라하고 사용자의 위치 데이터는 로 LBS Server에 Q=$\{p_u$, k} 형태로 전달된다. 본 논문에서 사용자의 질의 위치 프라이버시를 보호하는 것이 목적이므로, 는 공개키로 암호화되어 LBS Server에 전달된다.

LBS Server는 거리 계산을 위해 암호화된 D와 사용자의 위치 정보가 암호화된 Q를 수집한 상태이다.

LBS Server는 p_u와 D의 모든 위치 데이터 로 거리 계산을 수행하고, 계산된 결과를 저장하기 위해 LBS Server는 사전에 전송받은 공개키로 생성한 암호문을 생성한다. 연산으로 나온 결과는 LBS Server가 생성한 암호문 $Cipher_1, Cipher_2, ..., Cipher_d$에 각각 저장된다.

모든 거리 연산이 완료된 후, LBS Server는 결과가 저장된 암호문 $Cipher_1, Cipher_2, ..., Cipher_d$을 비교하여 오름차순으로 정렬한다. 이 때, 정렬 역시 암호문 상태에서 크기가 비교되며 정렬된다. LBS Server는 정렬된 암호문 중 사용자가 정의한 k개만큼을 결과 R=$\{Cipher_1, Cipher_2, ..., Cipher_k\}$로 사용자에게 반환한다.

2.2 기호설명

본 논문에서 쓰이는 주요 기호들은 다음과 같다.

- PK : 공개키 (Public Key)
- SK : 개인키 (Secret Key)
- $E_{PK}()$: 공개키로 데이터 암호화
- $D_{SK}()$: 거리 연산의 결과가 저장되어 있는 암호문을 개인키로 복호화
- Q : 사용자가 LBS Server로 전송하는 질의
- p_u : 사용자의 위치 데이터
- $f(D,Q)$: D와 Q로 암호문 상태로 LBS Server에서 연산되는 함수
- R : LBS Server가 사용자에게 결과로서 반환하는 암호문

2.3 Top-k 위치정보서비스 프로세스

Top-k 위치정보서비스 프로세스의 주체는 사용자, LBS Server, Database로 구성된다. 각 주체는 Top-k 위치정보서비스를 이용하기 전 암호화와 복호화를 위해 외부의 키 서버로부터 각각 PK 혹은 SK를 전송받아야한다. 외부의 키 서버는 PK와 SK를 사용자에 제공하고, LBS Server, Database에는 PK만을 전송한다. 키 전송이 완료되고, 각 주체는 해당하는 키를 갖고 있다는 것을 전제로 Top-k 위치정보서비스 프로세스는 진행된다. Fig. 1은 Top-k 위치정보서비스를 이용하는 과정을 나타냈다.

Top-k 위치정보서비스 과정에서 맨 처음 Database는 공개키로 위치 데이터 좌표 p_i를 각각 $E_{PK}()$한다. Database는 x좌표 y좌표로 나누어 암호화하고 암호화된 x좌표 y좌표를 한 쌍으로 구성한다. 즉, 암호화된 위치 데이터는 p_1 =($Cipher_{x1}$,$Cipher_{y1}$), p_2=($Cipher_{x2}$,$Cipher_{y2}$), ... , p_d =($Cipher_{xd}$,$Cipher_{yd}$)이고, 생성된 암호문은 암호화된 값으로 채워져 있는 상태이다. 각 위치 데이터가 공개키로 암호화되었으므로, 이는 $E_{PK}(p_1)$, $E_{PK}(p_2)$, ... , $E_{PK}(p_d)$로 정의된다. Database는 암호화된 일련의 데이터들의 집합 D를 LBS Server에 전송한다.

그 다음 사용자 역시 자신의 위치 정보 $p_u=(x_u,y_u)$

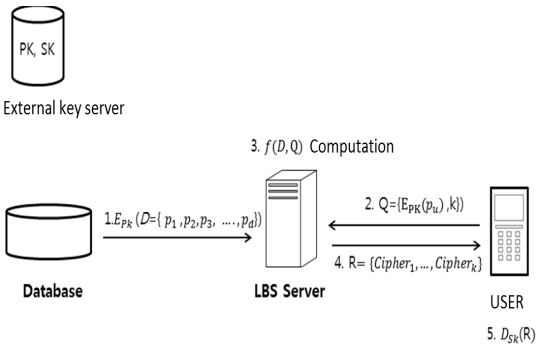

Fig. 1. Top-k LBS process

의 x좌표 y좌표를 각각 공개키로 암호화한다. 암호화된 사용자의 위치 정보는 p_u = ($Cipher_{xu}$, $Cipher_{yu}$)이며, 이는 Top-k 위치정보서비스 프로세스에서 $E_{PK}(p_u)$로 표현한다. 질의에는 사용자의 위치 정보 외에도 k가 포함되는데, 이 때 k는 사용자의 민감한 위치 정보가 아니라 LBS Server로부터 반환 받을 결과의 개수를 의미한다. 따라서 LBS Server에 질의로 전송될 k는 암호화될 필요가 없다. 사용자는 최종적으로 공개키로 암호화된 사용자의 위치 정보 $E_{PK}(p_u)$와 LBS Server가 참조할 파라미터 k가 포함된 질의 Q를 LBS Server에 전송한다.

모든 위치 데이터 전송이 완료된 후, LBS Server는 Database와 사용자로부터 전송받은 두 종류의 위치 데이터로 Top-k 연산을 수행한다. 이 때 $f(D,Q)$는 Top-k 위치정보 프로세스 중 LBS Server가 Top-k 연산을 수행하도록 사전에 정의된 함수이고, 아래는 함수 $f(D,Q)$의 알고리즘을 나타낸다.

입력

- Database의 암호화된 일련의 위치 데이터 D, 사용자의 질의 Q. (Q=암호화된 사용자의 위치 정보, 결과 반환 파라미터 k를 포함)

수행과정

1. LBS Server는 D의 데이터 집합과 Q에 포함된 위치 정보로 $f_{distance^2}()$ 연산을 수행. 이 때, $f_{distance^2}()$는 모든 데이터 객체와 사용자의 질의 위치 거리 연산을 암호문 상태에서 수행하여 결과를 암호문으로 반환하는 함수.
2. 함수 $f_{top-k}()$는 암호문 정렬과 상위 k개의 암호문을 결과로 반환하는 함수. LBS Server는 $f_{top-k}()$를 이용하여 암호문 상태의 $f_{distance^2}()$ 결과를 정렬하고 input에서 정의된 k개만큼의 암호문을 결과로서 사용자에게 전송.

출력 결과

- 사용자의 질의 위치로부터 가장 가까운 Top-k 위치 객체

D와 Q는 동일한 PK로 암호화되어 있기 때문에 LBS Server는 완전동형암호를 이용해 암호화된 상태에서 D와 Q로 거리 계산을 수행할 수 있다. LBS Server는 $f()$내에서 거리 연산의 결과가 저장되어 있는 암호문을 정렬한 후에, 정렬된 암호문 Cipher 중 상위 k개의 위치 데이터를 사용자에게 결과로 전달한다. 이 때, 결과 R={$Cipher_1$, $Cipher_2$, ... , $Cipher_k$}에는 Database로부터 전달받은 위치 데이터의 좌표 와 연산된 거리가 저장되어있다.

Top-k 위치정보서비스에서 사용자는 결과를 복호화할 수 있는 유일한 존재이다. 사용자는 사전에 SK를 외부의 키 서버로부터 획득했기 때문에 LBS Server가 전송한 결과 R을 SK로 복호화하여 평문을 확인할 수 있다. 복호화된 평문은 사용자의 위치로부터 가장 가까운 k개의 위치 좌표를 나타내므로, 사용자는 Top-k 위치정보서비스의 결과를 확인하고 서비스 이용을 종료한다.

2.4 Top-k 구현을 위한 세부 함수들

2.4.1 거리 연산 함수 $f_{distance^2}()$

거리 계산을 위해 D의 위치 데이터 중 하나를 (x_d, y_d)라고 하고, 사용자의 위치 정보를 포함한 Q의 위치 정보를 (x_u, y_u)라 가정한다. 이 때, LBS Server의 거리 계산는 다음과 같이 수행된다.

$$distance^2 = (x_p - x_u)^2 + (y_p - y_u)^2$$

원래의 거리 계산식은 제곱 덧셈의 결과에서 루트를 포함시켜야하지만, 본 논문에서는 단순 거리 비교를 위하여 루트를 제외하는 것으로 한다. 계산된 거리는 정수로 표현되며, 가장 작은 값의 결과가 나온 위치 데이터가 사용자의 위치로부터 가장 가까운 지점임을 나타낸다. 본 논문에서 $distance^2$은 LBS Server가 완전동형암호를 이용하여 연산을 수행할 함수 $f()$이다. 거리 연산 함수 $f_{distance^2}()$ 을 사용하기 위해 필요한 파라미터는 D와 Q이다. D는 사용자에게 위치 정보를 제공할 실질적인 위치 데이터이며, 이는 실제 지도 정보를 표현한다. 만약 D를 구성하는 위치 데이터가 6개 이하이고 사용자가 요청한 Top-k 결과로 6개의 위치 데이터를 전송해야한다면, LBS Server는 D 모두를 사용자에게 전송해야한다. 이는 D 자체를 사용자에게 노출하는 것과 같으므로, D는 충분히 큰 위치 데이터로 구성되어야한다.

Top-k 위치정보서비스를 이용하기 위해 사용자가 LBS Server에 질의한 Q는 LBS Server가 모든 데이터 D와 거리 연산을 완료할 때까지 변경되지 않고 사용된다. 거리 연산 함수 $f_{distance^2}()$은 아래의 알고리즘을 갖는다.

입력

- Database의 암호화된 일련의 위치 데이터 D, 사용자의 질의 Q. (Q=암호화된 사용자의 위치 정보, k(k=

정수))

수행과정

1. 암호문 상태에서 마이너스를 수행하는 함수 $f_{subtract}()$를 사용해 LBS Server는 D의 암호화된 위치 데이터 한 점 $(x_{p,}y_p)$과 암호화된 사용자의 위치 정보 $(x_{u,}y_u)$를 암호 상태에서 x좌표와 y좌표로 마이너스 $(x_p - x_u), (y_p - y_u)$.
2. 암호문 곱셈 연산을 수행하는 함수 $f_{multiplier}()$로 LBS Server는 $f_{subtract}()$의 결과를 암호화된 상태에서 제곱 계산 $(x_p - x_u)^2, (y_p - y_u)^2$.
3. 제곱 계산 연산된 결과 $(x_p - x_u)^2, (y_p - y_u)^2$를 암호화 상태에서 더하기 연산 수행 $(x_p - x_u)^2 + (y_p - y_u)^2$. 더하기 연산은 $f_{fadder}()$을 사용하여 연산.
4. LBS Server는 공개키로 새로운 암호문 Cipher 생성.
5. 암호화된 $(x_p - x_u)^2 + (y_p - y_u)^2$의 결과는 생성된 암호문 Cipher에 저장.
6. 위치 데이터 D가 모두 계산될 때까지 LBS Server는 과정 1번 ~ 5번을 반복.

출력 결과

• $distance^2$ 연산 결과를 저장한 암호문 d개

본 논문에서 다루는 Top-k 위치정보서비스에서 D와 Q는 암호화되었기 때문에 LBS Server에 어떠한 위치 데이터도 노출되지 않지만, 본 절의 설명을 위해 편의상 위치 좌표를 구성한다. Fig. 2는 D와 Q의 위치 좌표를 나타낸 것이다. D={}이고, 각 좌표의 위치는 =(2,4), =(1,2), =(1,7), =(6,8), =(5,6), =(6,3), =(5,1)으로 표시되었다. 질의에 포함된 사용자의 위치 정보는 Fig. 2에서 빨간 점으로 표시되었고, 좌표는 =(4,3)으로 정의했다.

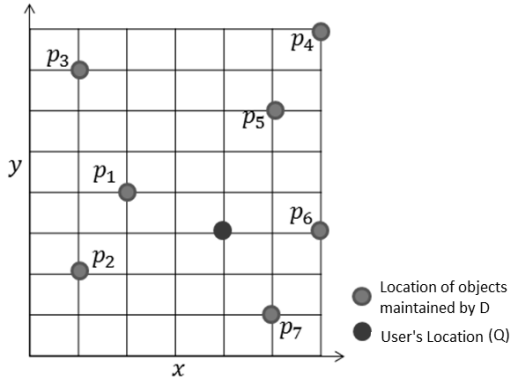

Fig. 2. An example of D and Q

LBS Server는 D와 Q로 암호문 상태의 연산을 수행하고, 수행된 연산의 결과는 LBS Server가 공개키로 생성한 암호문 Cipher에 저장된다. 예시로, Fig. 2에서 D의 좌표와 Q의 위치 정보 연산의 결과는 =5, =10, =25, =29, =10, =4, =5로서 새로 생성된 암호문에 각각 저장되어 반환된다. 이 때, LBS Server는 암호문에 저장된 연산 결과를 알지 못하고, 단지 함수 의 결과로서 반환된 암호문 Cipher 개수만 알 수 있다.

2.4.2 정렬 함수

LBS Server는 연산된 암호문 결과 값을 비교하여 상위 k를 사용자에게 반환해야한다. 암호문 상태에서 크기 비교를 수행하는 함수를 정렬 함수 라하고, 이는 거리 연산 함수 후에 LBS Server가 수행한다. 정렬 함수$f_{top-k}()$은 아래의 알고리즘을 갖는다.

입력

• 함수 $f_{distance^2}()$ 결과로 반환된 암호문 d개, 사용자가 정의한 파라미터 k

수행 과정

1. 암호문 d개를 하나의 배열에 저장.
2. 배열에 저장된 암호문을 암호화된 상태에서 두 개씩 비교. 암호문에 대응하는 평문의 값이 더 작은 암호문을 임의의 배열에 저장.
3. 임의의 배열에 저장된 d/2개의 암호문을 다시 두 개씩 비교 후 암호문에 해당하는 평문이 작은 값을 또 다른 임의의 배열에 저장.
4. 1번 ~ 4번까지의 과정이 d=1이 될 때까지 반복.
5. d=1로써 저장된 암호문은 top-1로 정의.
6. top-1로 반환된 암호문을 제외한 d-1개로 과정 1번 ~4번까지 암호문 상태의 비교를 반복.

출력 결과

• 상위 k개의 암호문 반환

정렬 함수 역시 암호문 상태에서 비교가 진행되기 때문에 LBS Server는 일체의 위치 정보에 대하여 알 수 없다. 하지만 본 절의 설명을 위하여 Fig. 2의 위치 좌표를

Table 1. The array of Top-k operation result with the points in Fig. 2

Points	$f(distance^2)$
$Cipher_6$	4
$Cipher_1$	5
$Cipher_7$	5
$Cipher_2$	10
$Cipher_5$	10
$Cipher_3$	25
$Cipher_4$	29

이용한 함수 결과를 Table 1처럼 배열에 저장하여 나타낸다. Table 1은 연산이 완료된 상태이며, 연산 결과가 저장되어 있는 암호문도 임의로 넘버링되어 배열에 정렬된 모습이다.

LBS Server는 정렬된 배열을 바탕으로 사용자가 원하는 k만큼의 Cipher를 결과 R={$Cipher_6$, $Cipher_1$, $Cipher_7$}을 반환한다. 결과를 반환 받은 사용자는 사전에 전송받은 SK를 사용해 R을 복호화하여 암호문에 대응하는 평문 결과를 확인할 수 있다.

3. 시뮬레이션을 통한 수행 시간 예측

3.1 환경 설정

본 논문에서 완전동형암호를 이용한 Top-k 위치정보 서비스의 유효성을 증명하고자 수행한 시뮬레이션 테스트 환경은 다음과 같이 구축되었다. 테스트 환경은 Ubuntu에서 Intel Xeon processor Quad-core 1600 MHz, 64GB RAM, HDD 916.8GB 하드웨어를 가지고 진행되었다.

LBS Server가 암호문 상태에서 거리 연산을 수행하고 결과를 올바로 사용자에게 반환해야하므로 실험에서 완전동형암호를 지원하는 라이브러리 HElib(Homomorphic Encryption Library)를 사용했다. HElib을 사용하기 위하여 gmp-6.0.0과 ntl-9.0.1 버전으로 세팅하였고 실험은 C++ 언어로 작성되었다.

또한 컴파일하기 전에 따로 설정해준 파라미터는 Recryption_size와 Key Generation이다. Recryption_size는 완전동형암호를 사용하면서 쌓이는 노이즈 한계치를 설정하여 암호 상태의 연산 과정 중에 노이즈를 한계치 이하로 줄여주기 위한 파라미터이다. 본 논문에서 Recryption_size는 -35로 정의하여 함수 연산 시 노이즈가 -35를 초과할 경우 부트스트래핑을 실행하였다. Key Generation는 암호화와 복호화에 필요한 공개키 쌍을 생성하기 위해 설정된 파라미터이고, 실험에서 이는 L=17, B=28로 설정하였다.

Table 2. Result of executing $f_{distance^2}()$

Query position	Location data in D	Distance computation result stored as encrypted data
p_u= (6, 7)	p_1=(25, 67)	3961
	p_2=(46, 44)	2969
	p_3=(21, 35)	1009
	p_4=(21, 66),	3706
	p_5=(32, 75)	5300
	p_6=(36, 86)	7141
	p_7=(11, 11)	41
	p_8=(19, 10)	178
	p_9=(53, 11)	2225
	p_{10}=(47, 64)	4930
	p_{11}=(38, 27)	1424
	p_{12}=(18, 26)	505
	p_{13}=(28, 47)	845
	p_{14}=(22, 22)	481
	p_{15}=(27, 11)	457
	p_{16}=(22, 11)	272

실험을 위해 사용된 Database의 위치 데이터 개수 d는 16으로 하였고, 각 위치 좌표는 D={=(25, 67), =(46, 44), =(21, 35), =(21, 66), =(32, 75), =(36, 86), =(11, 11), =(19, 10), =(53, 11), =(47, 64), =(38, 27), =(18, 26), =(28, 47), =(22, 22), =(27, 11), =(22, 13)}로 각각 암호화하였다. 사용자의 위치 정보는 = (6, 7)로 설정하여 암호문으로 만들었고, k는 3으로 하여 LBS Server에 질의하는 것으로 했다.

3.2 시뮬레이션 결과

암호화된 상태에서 D와 Q의 함수 의 결과는 Table 2와 같았다. Table 2는 편의상 암호문을 복호화하여 그에 해당하는 평문의 결과만을 나타내었다.

암호화된 상태에서 D의 16개의 각각의 좌표와 암호화된 사용자의 위치 정보로 함수 를 연산하고, 결과를 암호

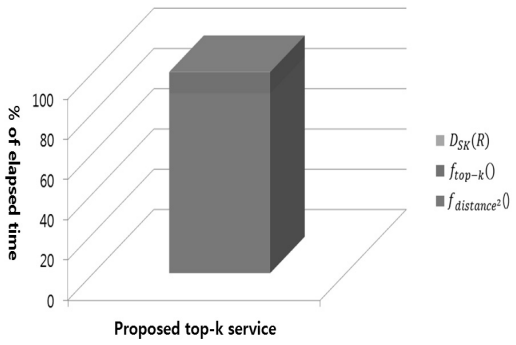

Fig. 3. Ratio of elapsed time among the core components in the proposed LBS system

화하여 반환하는 시간은 863472초가 걸렸다. 이는 연산 수행 시 노이즈 제거를 위한 부트스트래핑의 소요 시간까지 모두 포함한 결과이다.

암호문 상태에서 거리 연산이 완료된 후, LBS Server에서 사용자에게 가장 가까운 k의 위치를 반환하기 위하여 함수 를 실행한다. 비교 연산의 결과를 확인해 본 결과 오름차순으로 정렬된 것임을 확인할 수 있었다. 실제 동작 시스템에서는 LBS Server는 암호문에 해당하는 평문의 결과도, 결과가 저장되어 있는 암호문의 넘버링의 정보도 모르는 상태이다.

LBS Server는 함수 를 이용하여 암호문 정렬을 수행한다. 이 때 사용자가 결과로 반환할 파라미터 k를 3으로 설정하였으므로, 실험에 사용한 D의 위치 좌표 중 =(11, 11), =(19, 10), =(22, 11)가 차례대로 연산 결과 41, 178, 272를 가지며 사용자에게 결과로 반환된다. 함수 의 수행 시간은 100583초가 소요되어, LBS Server에서 완전동형암호를 이용한 Top-k 연산 및 결과 반환 시간은 총 963955초가 걸렸다.

또한 사용자가 LBS Server로부터 받은 암호문 결과를 복호화하는 시간은 매우 짧은 시간인 2.46573초가 소요되었다. 결국 사용자가 프라이버시를 보호하며 Top-k 위치정보서비스를 사용하는 시간은 약 963958초가 걸렸다. 이는 약 11일 5시간 18분 (약 270시간 정도)의 시간이다.

Fig. 3은 완전동형암호를 이용한 Top-k 위치정보서비스를 이용할 때 각 과정에서 걸리는 시간에 따른 백분율을 그래프로 나타낸다. Top-k 위치정보서비스 이용함에 있어 걸리는 약 963958초에서 함수 에 소요되는 시간은 약 89.44%이다. 나머지 시간은 함수 와 사용자의 결과 복호 시간은 총 소요 시간에서 10.56%만 차지하였다. 함수의 소요 시간이 큰 이유는 함수 의 결과를 반환하는데 많은 하위 함수의 과정을 거쳐야했기 때문이다. 또한 암호화 상태의 연산 과정에서 부트스트래핑의 횟수가 많았고 노이즈를 줄이는데 시간이 많이 소요되었다.

4. 결 론

본 연구에서는 완전동형암호를 이용한 Top-k 위치정보서비스를 제안하여 LBS Server에 노출되는 사용자의 위치 정보 프라이버시를 보호하는 방안을 제시하였다. 사용자의 위치 정보로부터 가장 가까운 k개의 암호문 상태 연산을 위해 함수 $f_{distance^2}()$를 사용하였고, 연산 결과도 암호문으로 반환되어 LBS Server에 결과가 노출되지 않도록 했다. 또한 LBS Server가 암호문의 연산 결과를 함수 $f_{top-k}()$로 암호문 상태에서 정렬하여 사용자에게 상위 k개를 결과로서 반환하도록 하였다. 제안 방법은 LBS Server에 어떠한 위치 정보도 노출되지 않아 완전동형암호를 이용한 Top-k 위치정보서비스를 이용하는 사용자의 프라이버시가 보호됨을 증명하였다.

본 연구의 제안 방법에서 사용자가 결과를 반환받기까지의 시간은 약 270시간으로 긴 시간이 소요되었지만, 이는 향후 하드웨어의 발전과 함께 개선될 수 있을 것이라 사료된다.

References

1. Nectaria Tryfona ,Dieter Pfoser. (2005). “Data Semantics in Location-based Services”, Journal on Data Semantics III Lecture Notes in Computer Science Volume 3534, p. 168-195.
2. Dieter Pfoser and Nectaria Tryfona. (2009). “The Use of Ontologies in Location-based Services: The Space and Time Ontology in Protégé”, Research Academic Computer Technology Institute, Athens, Greece.
3. Jaideep Vaidya and Chris Clifton. (2005). “Privacy-preserving top-k queries”, In Proc. ICDE.
4. Qian Chen, Haibo Hu, Jianliang Xu. (2013). “Authenticating top-k Queries in Location-based Services with Confidentiality”, VLDB Endowment.
5. Haibo Hu, Jianliang Xu, Qian Chen, Ziwei Yang. (2012). “Authenticating location-based services without compromising location privacy”, In Proc. SIGMOD.
6. T. Hashem and L. Kulik. (2011). “Don't trust anyone: Privacy protection for location-based services”, Journal of

Pervasive Mobile Computing, vol. 7, p.44-59.
7. Xiaoling Zhu, Yang Lu, Xiaojuan Zhu, Shuwei Qiu. (2013). “A Location Privacy-Preserving Protocol Based on Homomorphic Encryption and Key Agreement”, IEEE.
8. Shin, Kang Ju, Xiaoen Chen, Zhigang Hu, Xin. (2012). “Privacy protection for users of location-based services”, IEEE wireless communications.
9. Chi Lin, GuoweiWu1, Chang Wu Yu. (2014). “Protecting location privacy and query privacy a combinedclustering approach”, Lin_et_al-2014-Concurrency_and_Computation-_Practice_and_Experience.
10. P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. (2007). “Preventing Location-Based Identity Inference in Anonymous Spatial Queries”, IEEE TKDE, 19(12):1719-1733.
11. Sergey Yekhanin. (2010). “private information retrieval”, Communications of the ACM, Volume 53, Issue 4, p. 68-73.
12. Gabriel Ghinita, Panos Kalni, Ali Khoshgozara, Cyrus Shahab, Kian-Lee Ta. (2008). “Private queries in location based services: anonymizers are not necessary”, SIGMOD '08 Proceedings of the 2008 ACM SIGMOD international conference on Management of data, p. 121-132.

허 미 영 (jlee8099@naver.com)

2013 서울여자대학교 정보보호학 학사
2015 서울과학기술대학교 SW분석설계학과 석사
2015～현재 A3 Security 연구원

관심분야 : 침입탐지 분석, 응응 암호, 정보보호

이 윤 호 (younholee@seoultech.ac.kr)

2000 KAIST 전산학과 학사
2002 KAIST 전산학과 석사
2006 KAIST 전산학과 박사
2007 KAIST 정보전자연구소 박사후 연구원
2007～2009 GTISC 방문 박사후 연구원
2009～2013 영남대학교 정보통신공학과 조교수
2013～현재 서울과학기술대학교 글로벌융합산업공학과 교수

관심분야 : 응용 암호, 데이터 보안, 네트워크 보안