다. 이와 함께 이를 실제 서비스에 도입하였을 경

위치 기반 서비스의 보안 시스템

박찬현 · 이재흥 · 박용수

한양대학교 · 서울대학교 · 한양대학교

Security System for Location-Based Services

Chan Hyun Park · Jaeheung Lee · Yongsu Park

Hanyang University · Seoul National University · Hanyang University

E-mail : parkch0708@hanmail.net

요 약

위치 기반 서비스(Location-Based Services)란 사용자의 위치를 통하여 사용자가 필요한 정보를 전달해주는 서비스를 말한다. 최근 휴대용 스마트 기기(스마트폰, 태블릿 PC 등)를 사용하는 사람들이 급격하게 증가하면서 위치 기반 서비스의 이용률 역시 급격하게 증가하고 있다.위치 기반 서비스에서 가장 중요한 것 중 하나가 보안이다. 사용자의 개인 정보, 특히 질병 등의 민감한 내용의 정보가 노출 되어서는 안 된다. 본 논문에서는 위치 기반 서비스를 공격하는 방법의 예와 이를 방어하기 위한 보안 기술을 제시한다. 그리고 이를 실제 서비스에 도입하였을 경우 어떤 방식으로 적용할 수 있는지를 제시한다.

ABSTRACT

Location-Based Services(LBS) are a general class of computer program-level services used to include specific controls for location data as control features in computer programs. In recent years, the number of smart device(Smart Phone, Tablet PC etc.) users growth was exponential. For that reason, using rate of LBS has drastically increased. The most important thing of LBS is security. Personal information, especially private information likes illness, should not be disclosed. In this paper shows how to attack LBS and how to defense it.

키워드

위치 기반 서비스, LBS, 정보 보안, 스마트폰 보안

Ⅰ. 서 론

2010년 말부터 스마트 기기(스마트 폰, 태블릿 PC 등)의 보급률이 급격하게 증가하기 시작하였다. 이와 함께 위치 기반 서비스를 사용한 애플리케이션의 사용률 역시 증가하였다.

위치 기반 서비스를 사용하는 애플리케이션이 늘어가면서 이에 대한 정보 보안이 중요하게 되었다. 사용자의 위치 정보는 중요한 개인 정보 중 하나이다. 한 개인의 위치 정보가 유출 될 경우, 그 사람의 민감한 정보, 예를 들면 질병 정보나 생활 패턴 등이 노출 될 위험이 크다. [1-2]

본 논문에서는 위치 기반 서비스를 공격하는 방법이 어떤 것들이 있는지 소개하고, 이를 방어하여 정보 유출을 막기 위한 보안 기술을 제시한다. 이와 함께 이를 실제 서비스에 도입하였을 경우 어떤 방식으로 적용할 수 있는지를 제시한다.

Ⅱ. 본 론

2.1 위치 기반 서비스(LBS)

위치 기반 서비스(Location-Based Services, 이하 LBS)란 사용자의 위치 정보를 기반으로 사용자가 요청한 정보를 제공해 주는 서비스를 말한다. LBS를 사용한 대표적인 애플리케이션에는 지도, 네비게이션, 사진&앨범, 소셜 네트워크 서비스(SNS) 애플리케이션 등이 있다.

지도 앱이나 네비게이션 앱의 경우 LBS가 가장 핵심 기술이 되는 애플리케이션이라 할 수 있다. 애플 사(社)의 아이폰(iPhone)에 기본 애플리케이

션 중 하나인 사진 앱과 앨범 앱의 경우 LBS를 사용하여 사용자가 사진을 찍으면 사진을 찍은 위치 정보가 함께 저장되어 지도에서 사진을 찍은 위치를 볼 수 있다. 대표적인 SNS인 페이스북(Face Book)은 사용자가 글을 올릴 때 위치정보를 함께 올릴 수 있는 서비스를 제공한다.

2.2 LBS의 공격 방법

위치 기반 서비스를 제공하는 서버의 보안이 취약할 경우 이를 공격하여 얻은 정보를 바탕으로 개인 정보를 획득할 수 있다. 필명 서비스(Pseudonym Service)를 통하여 사용자가 누구인지를 직접적으로 나타내는 정보(이름, 주민등록 번호 등)를 숨기는 것은 가능하다. 하지만 사용자가 요청한 내용 안의 위치 정보만으로도 사용자가 누구인지 추측할 수 있다. [3]

Quasi-identifier

Age	ZipCode	Disease
42	25000	Flu
46	35000	AIDS
50	20000	Cancer
54	40000	Gastritis
48	50000	Dyspepsia
56	55000	Bronchitis

Name	Age	ZipCode
Andy	42	25000
Bill	46	35000
Ken	50	20000
Nash	54	40000
Mike	48	50000
Sam	56	55000

그림 1. 필명 서비스를 사용한 시스템을 공격하여 얻은 정보에서 사용자가 누구인지 추측하는 예시

Ⅲ. LBS 보안 시스템

3.1 LBS 보안 시스템 개요

위와 같이 위치 기반 서비스를 제공하는 서버가 공격당하여 정보가 유출 되었을 때 사용자가 누구인지를 추측하기 힘들게 하기 위하여 LBS서버에 사용자 개인의 위치 정보를 전송하지 않는다. 사용자 주변의 'k-1'명의 사람들과 사용자를 그룹화 하여, 총 k명의 사람들이 속한 그룹을 만든다. 사용자 개인의 위치 정보 대신 이 그룹에 속하는 사람들을 포함하는 범위를 전송함으로써 LBS를 제공하는 서버가 공격당하여 정보가 유출되어도 공격자가 특정 사용자를 알아낼 확률이 최대 1/k이 되도록 한다.

3.2 LBS 보안 시스템의 평가 기준

LBS 보안 시스템에서 가장 중요한 것은 익명성이다. 공격자가 LBS서버를 공격하여 얻은 정보 안에서 기존에 알고 있었던 정보 이외의 정보를 얻거나, 그룹화 되어 있는 모든 사용자(총 k명)의 위치 정보를 알고 있더라도 특정 사용자가 누구인지 1/k의 확률 이상으로 알 수 없어야 한다. [4] 또, 해당 그룹의 어떠한 사용자가 LBS를 이용 하더라도 똑같은 그룹을 생성하여 공격자가 k명 각각에 대한 공격이 모두 성공하여도 특정 사용자를 알아낼 수 없어야 한다.

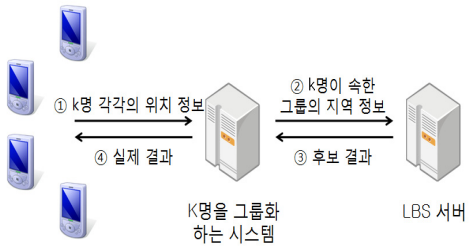

그림 2. LBS 보안 시스템 체계 [3]

Age	ZipCode	Disease
42-46	25000-35000	Flu
42-46	25000-35000	AIDS
50-54	20000-40000	Cancer
50-54	20000-40000	Gastritis
48-56	50000-55000	Dyspepsia
48-56	50000-55000	Bronchitis

Name	Age	ZipCode
Andy	42	25000
Bill	46	35000
Ken	50	20000
Nash	54	40000
Mike	48	50000
Sam	56	55000

그림 3. K=2일 때, 공격자가 특정 사용자를 알아낼 확률이 1/2임을 보여주는 예시

그리고 k명의 사용자를 포함하는 범위를 최대한 작게 함으로써 사용자에게 최대한 정확한 정보를 전달할 수 있어야 한다. LBS 서버에 전송하는 범위가 커질수록 정확도는 떨어진다고 볼 수 있다.

또 k명의 사용자 그룹을 빠르게 생성하여 사용자에게 조금 더 빠르게 서비스가 제공될 수 있도록 하여야 한다. 생성뿐만 아니라 사용자 그룹에 새로운 사용자를 추가하거나, 기존의 사용자를 삭제할 때에도 빠른 처리가 이루어 져야 한다.

3.3 제안 기법 알고리즘 개요

Neighbor Heuristic 기법을 사용하여 서비스를 요청하는 사용자를 포함한 사용자 주변 n명의 사람을 트리 구조로 그룹 화한다. 이 때 n은 k보다 큰 수로 설정하도록 한다. 이 때 트리를 생성하는데 걸리는 시간은 O(N^2)이 된다.

트리를 생성한 후 서비스를 요청한 사용자를 기준으로 최초로 k명 이상이 되는 부분 트리를 찾는다. 그 부분 트리 안에 모든 사용자를 포괄하는 지역을 LBS 서버로 전송한다.

3.3.1 트리 생성 알고리즘

Neighbor Heuristic 기법을 사용하여 n명의 사용자를 포함한 트리를 생성하는 알고리즘은 다음과 같다.

1. n명의 사용자를 각 클러스터 당 1명씩만 포함되도록 n개의 클러스터로 분할.
2. 모든 클러스터 사이의 거리 계산
3. 가장 가까운 거리의 두 클러스터를 병합
4. 3에서 병합된 클러스터와 다른 클러스터

사이의 거리를 계산.

5. n명의 사용자 모두를 포함한 1개의 클러스터가 생성될 때 까지 3번으로 돌아가 반복.

3번에서 두 클러스터를 병합할 때 사용되는 클러스터 병합 알고리즘은 다음과 같다.

1. 가장 작은 사용자 아이디를 가지는 클러스터의 트리가 왼쪽 서브트리가 될 수 있도록 이동한다.
2. 다음의 거리를 계산하고, 최소값을 찾는다.

① d1 = d(왼쪽 서브 트리의 가장 왼쪽 사용자, 오른쪽 서브 트리의 가장 왼쪽 사용자)
② d2 = d(왼쪽 서브 트리의 가장 왼쪽 사용자, 오른쪽 서브 트리의 가장 오른쪽 사용자)
③ d3 = d(왼쪽 서브 트리의 가장 오른쪽 사용자, 오른쪽 서브 트리의 가장 왼쪽 사용자)
④ d4 = d(왼쪽 서브 트리의 가장 오른쪽 사용자, 오른쪽 서브 트리의 가장 오른쪽 사용자)

3. 실제 거리가 가까운 사용자들끼리 트리 내부에서도 서로 가까이 있도록 각 서브 트리를 상황에 맞게 순서를 유지시키면서 flipping.

① 최소값이 d1일 때 – 왼쪽 서브 트리.
② 최소값이 d2일 때 – 양쪽 서브 트리.
③ 최소값이 d3일 때 – flipping 하지 않음.
④ 최소값이 d4일 때 – 오른쪽 서브 트리.

3.3.2 LBS 서버로 부분 트리를 전송

3.3.1에서 n명의 사용자를 트리 구조화 하였다면, 이 트리에서 서비스를 요청한 사용자를 기준으로 최초로 k명 이상이 되는 부분 트리를 찾아 이를 LBS 서버로 전송한다. LBS 서버로 전송하는 부분 트리를 찾는 알고리즘은 다음과 같다

1. 서비스를 요청한 사용자의 노드를 부분 트리의 루트 노드로 둔다.
2. 루트 노드 하위의 모든 노드의 개수를 카운팅 한다.
3. 만약 2번에서 구한 수가 'k−1'보다 작을 경우, 루트 노드의 부모 노드를 부분 트리의 새로운 루트 노드로 설정한다.
4. 2번에서 구한 수가 'k−1' 이상이 될 때까지, 즉 부분 트리의 모든 노드의 개수가 최소 k개가 될 때 까지 2번으로 돌아가 반복한다.

위와 같은 방법으로 서비스를 요청한 사용자를 포함하여 최소 k명으로 구성된 부분 트리를 LBS 서버로 전송하여 사용자가 서비스를 받을 수 있도록 한다.

Ⅳ.애플리케이션에 적용

4.1 LBS를 사용하는 애플리케이션에 적용

LBS를 사용하는 애플리케이션을 제작할 때 본 알고리즘을 그대로 사용하는 데는 몇 가지 문제가 있을 수 있다. 애플리케이션의 종류에 따라 그 문제점과 해결 방안이 다를 수 있다. 때문에 LBS를 사용하는 가장 보편적인 애플리케이션인 지도 애플리케이션을 예로 들도록 하겠다.

보통 지도 애플리케이션의 기능은 크게 두 가지가 있다. 하나는 현재 위치에서 특정 장소로 가는 경로를 알려주는 것 이고(서비스 A), 또 다른 하나는 사용자가 특정 장소가 아닌, 여러 장소를 포함하는 단어를 입력하면 주변에 해당 단어에 적합한 장소를 가까운 순서대로 표시해 주는 것(서비스 B) 이다. 사용자가 '편의점'이라는 단어를 입력하였을 때, 사용자 주변의 편의점을 가까운 순서대로 일정 기준에 맞게 알려주는 것이 서비스 B의 예라고 할 수 있다.

사용자가 서비스 A를 사용할 경우, 사용자가 서비스를 요청하면 우선 사용자를 사용자 주변의 다른 최소 k−1명의 사람들과 함께 그룹화 할 것이다. 그리고 이 그룹을 대표하는 위치 정보를 서비스를 제공하는 서버로 전송하여 이 그룹의 위치에서 특정 목적지 까지 가는 경로를 전송해줄 것이다. 여기에 서비스를 요청한 사용자의 위치에서 그룹을 대표하는 위치까지의 경로를 더하여 사용자에게 특정 목적지까지의 경로를 알려 줄 것이다.

서비스 B를 사용할 경우 조금 복잡할 수 있다. 그룹화 하여 그룹의 대표 위치 정보를 서버에 전송하여 주변의 해당 장소의 위치 정보들을 받는 것 까지는 서비스 A와 동일하다. 하지만 서비스 B에는 가까운 순서대로 나열해 주는 것이 표함되어 있다. 때문에 서비스를 요청한 사용자 개인의 위치에 따라 가까운 순서가 다를 가능성이 있으므로 이를 한 번 더 계산해야 한다.

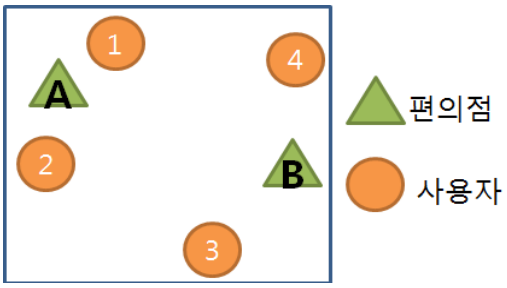

그림 4. 서비스를 요청한 사용자의 실제 위치에 따라 가까운 순서가 다를 수 있음을 보여주는 예시

<그림 4>는 k=4일 때, 사용자가 주변의 편의점을 검색하고 싶어 '편의점'을 검색하였을 때의

예시이다. 사용자가 서비스를 요청하면 우선 사용자와 주변의 3명의 사용자를 그룹화 할 것이다. 그리고 그룹을 대표하는 위치 정보를LBS 서버에 전송하여 주변의 편의점을 검색할 것이다. 편의상 그룹의 대표 위치에서 가장 가까운 편의점이 A, 두 번째로 가까운 편의점을 B라고 하겠다. 사용자 1과 사용자 2의 경우, 사용자와 가장 가까운 편의점이 A이고, 두 번째로 가까운 편의점이 B여서 LBS 서버에서 전송해 준 결과와 일치하지만, 사용자 3과 사용자 4의 경우 결과가 이와 반대이다. 이러한 이유 때문에 한 번 더 거리 계산을 하여 사용자가 정확한 결과를 얻을 수 있도록 해야 한다.

4.2 지도 애플리케이션에 적용할 때의 문제점

본 기술의 핵심은 서비스를 제공하는 서버에 서비스를 요청하는 사용자 개인의 위치 정보만 전송하는 것이 아닌, 사용자와 사용자 주변의 다른 사용자들을 그룹화 하여 이 그룹의 위치 정보를 전송하여 공격자가 공격에 성공하여도 특정한 사용자를 찾아낼 확률을 줄이는 데 있었다. 하지만 지도 애플리케이션을 제작할 때 이 방식을 사용하면 발생할 수 있는 문제가 있다.

k명의 사람들을 포함하는 지역의 크기가 커지게 되면 익명성의 문제가 생겨버린다. 이 문제는 k명의 사용자들이 서비스 B를 이용하고, 동일한 단어를 입력 하였을 때 발생할 수 있다. <그림 4>에서 볼 수 있듯이 사용자의 실제 위치에 따라 가까운 순서가 다를 수 있다. 이 확률은 k명의 사람들을 포함하는 지역의 크기가 커질수록 높아진다. 때문에 k명의 사용자 전원이 동일한 단어를 입력하여 서비스를 사용하였고, 공격자가 사용자 k명 전원을 공격하여 사용자가 받은 정보, 예를 들어 가장 가까운 편의점이 어디인지를 알아내었고, 그 결과가 동일하지 않다면, 공격자가 특정 사용자를 알아낼 확률이 1/k보다 커지게 된다.

4.3 문제점 해결 방안

위와 같은 문제를 해결하기 위해서는 k명의 사용자들을 포함하는 그룹의 대표 위치 정보를 1개로 하고, 사용자가 서비스 B를 사용할 때의 과정 중 제일 마지막 단계의 계산을 애플리케이션을 구동하는 사용자 개개인의 스마트 기기에서 하면 된다.

물론 이렇게 한다면 사용자 개개인의 스마트 기기가 처리해야할 계산이 많아져 처리 시간이 길어지거나 기기의 메모리 사용량이 늘어날 수는 있다. 하지만 본 기술의 최우선 목적이 익명성 즉 보안성이기 때문에 이는 감수해야할 부분이라 할 수 있다.

V. 결 론

지금까지 위치 기반 서비스(LBS)가 무엇인지, 공격자가 어떠한 방법으로 LBS를 공격하는지 설명하였다. 그리고 이를 방어하기 위한 시스템을 제안하였고, 이를 LBS를 사용하는 가장 보편적인 애플리케이션인 지도 애플리케이션에 적용하였을 때 발생하는 문제점과 해결책을 제시하였다.

그룹의 사용자들을 트리 구조화 하는데 소요되는 시간이 O(N^2)으로 조금 오래 걸린다는 단점이 있다. 그리고 이 기술을 실제 LBS를 사용하는 애플리케이션에 적용할 경우, 애플리케이션을 구동하는 스마트 기기에서의 처리 작업이 어느 정도 필요하다.

본 기술을 각 애플리케이션에 잘 적용시켜 기기에서의 처리 작업을 최소화 하여 전체 소요 시간을 줄일 수 있다면, 사용자의 개인 정보를 보호하는데 큰 도움을 줄 것이다.

참고문헌

[1] Alastair R. Beresford and Frank Stajano, “Location Privacy in Pervasive Computing”, IEEE Pervasive Computing, Vol.2, No.1, pp.46-55, 2003

[2] Claudio Bettini, Xiaoyang Sean Wang and Sushil Jajodia, “Protecting Privacy against Location-Based Personal Identification”, Proc. of SDM 2005, Trondheim, Norway, September, pp.185-199, 2005

[3] Panos Kalnis, Gabriel Ghinita, K. Mouratidis and Dimitris Papadias, "Preventing Location-Based Identity Inference in Anonymous Spatial Queries", IEEE Transactions on Knowledge and Data Engineering, Vol.19, No.12, pp.1719-1733, 2007

[4] Bugra Gedik and Ling Liu, "A Customizable K-Anonymity Model for Protecting Location Privacy", Proceedings of the IEEE International Conference on Distributed Computing Systems, Tokyo, japan, March, pp.620-629, 2004