Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.24, NO.1, Feb. 2014 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2014.24.1.59

소셜 네트워크 서비스를 위한 선별적 암호화 기능을 제공하는 관계 기반 동적 접근제어 모델*

권 근,† 정 영 만, 정 재 욱, 최 윤 성, 전 웅 렬, 원 동 호‡
성균관대학교

Relationship-based Dynamic Access Control Model with Choosable Encryption for Social Network Service*

Keun Kwon,† Youngman Jung, Jaewook Jung, Younsung Choi,
Woongryul Jeon, Dongho Won‡
Sungkyunkwan University

요 약

소셜 네트워크 서비스는 온라인상에서 정보의 공개성과 관계의 확장성을 기반으로 사용자의 개성을 표현하고 인적 네트워크를 강화시켜 주는 서비스이다. 하지만 이러한 특성은 사용자의 개인정보를 확산시키고 신뢰할 수 없는 정보에 접근하도록 하는 부작용을 초래한다. 따라서 사용자의 프라이버시를 보호하기 위한 다양한 접근제어 기법이 제안되었다. 하지만 데이터 암호화 접근제어 기법은 사용자와 직접적인 관계를 맺은 대상에 대한 접근제어만 지원하며 데이터 비 암호화 접근제어 기법은 서비스 제공자가 모든 정보를 열람할 수 있게 한다. 또한 두 접근제어 기법 모두 사용자 신뢰도에 대한 동적인 변화를 고려하지 않아 정적 접근제어만 제공하는 문제점이 있다. 이에 본 논문에서는 사용자가 선별한 민감한 데이터에 대한 암호화를 제공하는 관계 기반 동적 접근제어 모델을 제안하여 소셜 네트워크 서비스의 특성에 적합하면서 프라이버시 침해에 대한 보안성을 향상시킬 수 있는 방법을 제공한다.

ABSTRACT

The social network service is a online service letting users express the personality and enhancing the human network. However, these features result in side effects which diffuse personal information and make users access to treacherous information. Therefore, various access control models have been proposed. However, the access control mechanisms which encrypt data are only able to be applied for controlling access from direct node, and the access control mechanisms without data encryption allow service provider to access all the information. Moreover, both mechanisms do not consider dynamic changes in reliability of the users. In this paper, we propose relationship-based dynamic access control model including encryption of sensitive data, which consider the characteristics of SNS and improves the security of SNS.

Keywords: Social Network Service, Privacy, Access control, Reliability evaluation, Proxy re-encryption

접수일(2014년 1월 14일), 수정일(2014년 2월 17일),
게재확정일(2014년 2월 17일)
* 본 연구는 미래부가 지원한 2013년 정보통신·방송(ICT) 연구개발사업의 연구결과로 수행되었음.

† 주저자, kkwon@security.re.kr
‡ 교신저자, dhwon@security.re.kr (Corresponding author)

I. 서 론

소셜 네트워크 서비스(Social Network Service, 이하 SNS)는 온라인상에서 개인 중심의 정보 생성과 공유를 기반으로 사용자 간의 인적 네트워크를 형성하고 유지하며 확장할 수 있도록 해주는 서비스이다. SNS는 사용자로 하여금 자신의 신상 프로필 정보를 공개, 또는 반공개적으로 게시할 수 있게 하며, 관계를 맺은 다른 사용자들의 목록을 제공한다. 그리고 각 사용자의 인적 관계 목록을 서로 공유할 수 있도록 해준다[1]. 사용자들은 이와 같은 SNS의 서비스 기능을 바탕으로 자신의 신상 프로필 정보와 함께 개성을 표현하는 자료들을 게시하며 다른 사용자들과 이를 공유하고 소통하기 위해서 온라인상의 인적 관계를 맺게 된다. 이렇게 형성된 관계는 SNS의 인적 관계 공유 기능을 통해 더 강화되고 확대되어 오프라인상에서의 인적 네트워크보다 더 개방되고 확장된 온라인상의 대규모 인적 네트워크를 형성한다[2].

하지만 SNS가 가지는 개인정보의 공개성, 인적 관계의 확장성과 같은 특성은 사용자의 민감한 개인정보를 무분별하게 노출시키고 사용자로 하여금 신뢰할 수 없는 정보를 열람하도록 유도하여 다양한 부작용을 초래한다. 이러한 부작용 중 가장 논란이 되는 것은 사용자의 프라이버시 침해 문제이다[3][4]. SNS는 신상 정보와 개인적인 자료의 교환을 근간으로 구성되므로 프라이버시 침해와 직결될 수밖에 없으며 한번 노출된 개인정보는 정보의 전파속도가 빠른 SNS의 특성으로 인해 순식간에 확산된다[5]. 그리고 악의적인 목적으로 대량 수집되는 개인정보의 잠재적 남용 문제는 가장 심각한 부작용이라고 할 수 있다[6]. 또한 신뢰할 수 없는 정보의 확산은 개인의 문제를 넘어서 사회 전체에 영향을 미칠 수 있다[7][8].

SNS의 특성으로 인해 발생하는 이러한 부작용을 해결하기 위해 다양한 연구들이 진행되어오고 있으며 특히 프라이버시 침해 문제를 해결하기 위해서 각 사용자들 사이에서 데이터에 대한 접근을 제어하는 접근제어 기법들이 제안되었다.

SNS 사용자의 프라이버시 보호를 위해 제안된 접근제어 기법은 크게 두 가지로 분류할 수 있다. 첫 번째는 데이터 암호화 접근제어 기법으로 SNS 상의 데이터를 모두 암호화 하고 사용자들을 그룹으로 구분하여 특정 그룹에 소속된 멤버에게만 암호화된 데이터를 복호화 할 수 있는 키를 분배하는 방법이다[9][10][11]. 이 방식에서는 데이터 암호화 시에 동적으로 그룹을 형성하여 해당 그룹 멤버에게만 데이터에 대한 접근을 허용한다. 두 번째는 데이터 비 암호화 접근제어 기법으로 SNS 시스템, 또는 SNS 사용자가 수립한 정책에 따라 데이터에 대한 타 사용자의 접근을 제어하는 방법이다[12][13][14]. 이 방식에서는 주로 사용자간에 맺고 있는 관계를 기반으로 정책을 수립하여 접근제어에 적용한다. 하지만 데이터 암호화 접근제어 기법은 데이터 암호화 시에 자신과 직접적으로 관계를 맺은 사용자만을 대상으로 동적 그룹 멤버가 결정되며 평문 데이터에 대한 접근제어를 지원하지 않는다. 따라서 직접적인 관계를 맺지 않은 사용자에 대한 세분화된 접근제어(fine-grained access control)가 불가능하여 SNS의 특성인 관계의 확장성에 부합하지 못하는 문제점이 있다. 그리고 SNS상에 저장되는 모든 데이터를 암호화해야 하고 그룹 멤버의 탈퇴에 따른 키 폐기와 재분배를 수행해야 하므로 사용자에게 과도한 연산 부하를 요구한다. 반면에 데이터 비 암호화 접근제어 기법은 모든 데이터가 평문형태로 저장되고 사용자가 수립한 정책에 의한 접근제어 수행은 서비스 제공자가 실시하므로 사용자에게 과도한 연산부하를 요구하지 않는다. 하지만 서비스 제공자가 사용자의 민감한 개인정보를 포함하는 데이터에 접근할 수 있게 하는 단점을 가진다. 또한 두 접근제어 기법 모두 사용자 사이의 관계 강도를 나타내는 지표인 사용자 신뢰도와 신뢰도의 동적인 변화를 고려하지 않아서 동일 그룹 멤버에 대한 차별적 접근제어가 어렵고 한번 인가된 사용자에게 지속적으로 접근을 허가하는 정적 접근제어만을 제공하는 문제점이 있다[15].

이에 본 논문에서는 관계기반 접근제어 정책 모델[14]에 사용자의 동적 신뢰도를 평가하는 방법[16]을 결합시키고 사용자 스스로 민감한 데이터를 선별해서 암호화하는 기능을 추가하여 외부 사용자와 서비스 제공자로부터의 프라이버시 침해를 방지하고 SNS의 특성에도 적합한 접근제어 기법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 SNS에서의 접근제어 기법에 대한 요구사항을 도출하고 기존에 제안된 접근제어 기법의 장단점을 분석한다. 3장에서는 신뢰도 평가 기법과 데이터의 선별적 암호화 기능이 포함된 접근제어 기법을 제안한다. 그리고 4장에서는 기존에 제안된 접근제어 기법과 본 논문이 제안하는 접근제어 기법을 비교 분석하며 5장에서는 본 논문에 대한 결론을 맺는다.

II. 관련 연구

본 장에서는 SNS의 특성을 고려하여 SNS에 적합한 접근제어 기법의 요구사항을 도출한다. 그리고 기존에 제안된 접근제어 기법들의 장단점을 분석하여 도출된 요구사항 대한 적합성을 검토한다.

2.1 SNS를 위한 접근제어 요구사항

SNS에 적용하기 위한 접근제어 기법은 SNS가 제공하는 서비스 기능들의 동작을 최대한 보장하면서 동시에 사용자의 프라이버시를 보호할 수 있어야 한다. 이에 이러한 조건을 만족시키는 다양한 접근제어 요구사항이 도출되었으며 본 논문에서는 기존 연구에서 도출된 요구사항[17][18]에 사용자 신뢰도와 신뢰도의 동적인 변화, 서비스 제공자로 부터의 프라이버시 침해 가능성을 고려하여 아래와 같은 SNS 접근제어 요구사항을 도출하였다.

2.1.1 접근제어 정책의 개별화 지원

SNS에서의 접근제어 정책은 특정 자원(resource)을 소유하고 있는 사용자 스스로 수립할 수 있어야 한다. 그리고 해당 자원에 대한 소유권이 없는 사용자도 그 자원에 직접적인 관련이 있다면(e.g., 사진에 태그된 사용자) 접근제어에 반영될 자신의 정책을 개별적으로 수립할 수 있어야 한다. 또한 자원에 접근하고자 하는 사용자도 자신에게 적합하지 않은 자원으로의 접근을 방지하기 위한 정책을 수립할 수 있어야 한다. 이러한 개별적인 정책들이 한 자원에 대한 접근인가 여부를 결정할 때 동시에 적용되어야 한다.

2.1.2 사용자 타깃 행위에 대한 접근제어 지원

일반적인 접근제어 모델은 사용자가 소유한 데이터 자원을 타깃으로 하는 행위들(e.g, 읽기, 쓰기, 수정 등)에 대한 정책 수립만 지원한다. 하지만 SNS에서는 각 사용자들끼리 자원으로의 접근 없이 직접적으로 수행하는 행위들(e.g, 친구 추천, '좋아요' 클릭 등)이 빈번하게 발생한다. 따라서 이와 같은 사용자 타깃 행위들에 대한 접근제어를 지원해야 한다.

2.1.3 수신행위와 발신행위에 대한 정책 수립 지원

SNS에서 발생하는 행위는 정책 수립 사용자의 관점에 따라 두 종류로 구분할 수 있다. 첫 번째는 정책 수립 사용자, 혹은 소유 자원을 타깃으로 다른 사용자가 접근주체가 되어 수행하는 수신행위(incoming action)이다. 그리고 두 번째는 정책 수립 사용자가 접근주체가 되어 다른 사용자, 혹은 자원을 타깃으로 수행하는 발신행위(outgoing action)이다. SNS상에서의 모든 접근에서 수신행위와 발신행위는 동시에 발생하므로 정책 수립 사용자는 두 행위에 대한 정책을 수립할 수 있어야 한다.

2.1.4 간접 연결 사용자에 대한 접근제어 지원

SNS 사용자들은 개인정보의 공개와 인적 관계 맺기, 인적 관계 목록의 공유를 통해 보다 확장된 관계를 형성하기를 원한다. 그러므로 SNS를 위한 접근제어 기법은 자신과 이미 직접적인 관계를 맺은 직접 연결 사용자(direct friend)뿐만 아니라 아직 관계를 형성하지 않은 간접 연결 사용자(indirect friend)에 대해서도 무조건적인 접근 제한이 아닌 세분화된 접근제어를 수행할 수 있어야 한다.

2.1.5 신뢰도의 동적 변화를 반영한 접근제어 지원

SNS에서는 동일한 유형의 관계가 형성된 사용자 사이에서도 상호 교류 정도나 가치 공유 정도에 따라서 관계의 강도가 달라진다. 따라서 이러한 관계의 강도를 수치화한 사용자간 신뢰도가 접근인가의 결정에 영향을 미쳐야 한다. 또한 사용자 신뢰도는 동적으로 변화하는 특성을 가지고 있으므로 접근제어 기법은 한 번 접근이 인가된 사용자일지라도 신뢰도 값의 변화에 따라 접근 인가의 결정을 변경할 수 있어야 한다. 이는 동일 그룹 멤버나 동일한 관계를 형성하고 있는 사용자에 대해서 좀 더 세분화된 접근제어를 수행할 수 있도록 해준다.

2.1.6 민감한 데이터에 대한 암호화 지원

상용화된 SNS에서 사용자 데이터는 서비스 제공자의 서버에 저장되어 타깃 광고와 같은 수익사업에 활용될 수 있다[4]. 따라서 민감한 개인정보에 대한 암호화를 수행하여 서비스 제공자로부터의 프라이버시 침해를 방지해야 한다. 이 때 적용되는 암호화 기법은 사용자에게 과도한 연산부하를 요구하지 않아야 하며 기존 관계의 종료에 따른 키 갱신 및 재 암호화

에 대한 연산 부하를 최소화해야 한다.

2.2 데이터 암호화 접근제어 기법

시스템상의 모든 데이터를 암호화하고 복호화 키의 분배를 통해 데이터에 접근할 수 있는 권한을 부여하는 것은 민감한 데이터에 대한 기밀성을 보장할 수 있는 접근제어 방식이다. 이에 SNS 시스템에서 사용자 데이터에 대한 암호화를 수행하는 다양한 프라이버시 보호 기법들이 제안되었다.

Saikat Guha 등은 [19]에서 사용자 프로필 정보를 의사난수 환자 암호(pseudorandom substitution cipher)와 대칭키 암호를 사용해서 다른 사용자의 프로필 정보로 대체하는 기법인 NOYB를 제안하였다. 이 기법은 프로필 게시자의 대칭키를 분배받은 사용자만이 게시자의 실제 프로필을 복호화 할 수 있게 하여 프라이버시를 보호한다. 하지만 오직 사용자 프로필에 대한 보호만 가능하다는 한계점을 가지며 직접 연결 사용자에 대한 그룹화를 지원하지 않으므로 세분화된 접근제어가 불가능하다. 또한 관계의 종료가 발생하면 대칭키의 폐기(revocation)를 위해서 모든 사용자에게 새로운 키를 재분배해야 한다.

Matthew M. Lucas 등은 [20]에서 사용자 프로필뿐만 아니라 일반 게시물에 대한 프라이버시도 보호해주는 외부어플리케이션 형태의flyByNight를 제안하였다. 이 기법에서 일대일(one-to-one) 전송되는 메시지는 엘가말 암호(ElGamal Scheme)를 사용해서 암호화 된다. 그리고 사용자 그룹을 대상으로 하는 일대다(one-to-many) 전송 메시지에 엘가말 암호를 사용한 프록시 재 암호화(proxy re-encryption)기법을 적용하여 메시지 송신자의 연산효율성과 키 저장 공간 효율성을 향상 시킨다. 하지만 여러 그룹에 동시에 멤버로 등록된 사용자에 대한 선별적 메시지 암호화가 어렵고 기존 멤버를 삭제 할 경우 모든 키를 갱신해야 하는 문제점이 있다.

Randy Baden 등은 [9]에서 기존의 기법보다 세분화된 접근제어를 제공하기 위해서 전통적인 공개키 암호와 속성 기반 암호(attribute-based encryption)을 함께 적용한 Persona를 제안하였다. 이 기법은 직접 연결 사용자를 속성(attribute)에 대응하는 그룹으로 구분하고 해당 속성이 포함된 복호화 키를 각 사용자에게 분배한다. 그리고 암호화 시에 속성들의 논리적 조합을 접근 구조(access structure) 형태로 암호문에 포함시켜서 동적으로 그룹을 형성한다. 복호화는 접근 구조를 만족시키는 속성이 포함된 복호화 키를 가지고 있는 그룹 멤버만 수행할 수 있으므로 여러 그룹에 동시에 소속된 멤버들에 대한 선별적 접근제어가 가능한 장점이 있다. 또한 ACL(Access Control List)을 통해 사용자와 관련된 비 데이터 자원에 접근하는 행위에 대한 제어도 가능하다. 하지만 데이터를 암호화 하는 기존의 기법들과 마찬가지로 간접 연결 사용자에 대한 접근제어를 지원하지 못하는 한계점이 있다. 그리고 그룹멤버 탈퇴에 따른 키 폐기를 위해 나머지 그룹 멤버에게 새로운 키를 분배해야 하며 기존에 암호화된 데이터에 대한 재 암호화를 수행해야 하는 단점이 있다. 이에 persona 이후에 제안된 데이터 암호화 접근제어 기법들은 키 폐기 효율성을 개선하기 위한 기법들이 주를 이룬다.

Sonia Jahid 등은 [10]에서 속성 기반 암호를 사용한 접근제어 기법에 프록시(proxy)를 적용하여 키 폐기 효율성을 향상시킨 EASiER를 제안하였다. 이 기법에서 복호화 키에 포함되는 속성은 Shamir의 비밀 분산법(secret sharing)[21]에 의해 블라인드 되며 프록시는 암호문을 변형하여 탈퇴되지 않은 멤버만 복호화 키의 속성을 복구해 낼 수 있도록 해준다. 따라서 탈퇴된 멤버는 암호화된 데이터를 더 이상 복호화 할 수 없으며 키 폐기를 위한 추가적인 키 분배나 기존 데이터의 재 암호화 과정이 요구되지 않는다. 하지만 이 기법 데이터를 타깃으로 하는 행위에 대한 접근제어만 수행하는 한계점이 있으며 간접 연결 사용자에 대한 접근제어를 제공하지 않는다.

Fatemeh Raji 등은 [11]에서 속성 기반 암호 대신에 IBBE(ID-based Broadcast Encryption)을 적용한 접근제어 기법을 제안하였다. 이 기법에서 메시지 송신자는 자신이 원하는 직접 연결 사용자만 선별하여 그룹을 형성하고 해당 멤버들의 ID만 선택하여 IBBE를 통해 헤더와 데이터 암호화 키를 생성한다. 이 키는 AES(Advanced Encryption Standard)같은 대칭키 암호 알고리즘의 키로 사용되어 데이터를 암호화 하며 메시지 수신자는 자신의 ID가 포함된 헤더로부터 데이터 암호화 키를 유도해 내어 데이터를 복호화 할 수 있다. 이 기법에서는 탈퇴 멤버의 ID만 삭제하면 키 폐기와 동일한 효과를 얻으므로 효율적인 그룹 멤버 변경이 가능하다. 하지만 EASiER와 마찬가지로 데이터만 고려하므로 사용자 타깃 행위에 대한 접근제어를 지원하지 않으며 간접 연결 사용자로부터의 접근을 무조건적으로 제한하는 한계점을 가진다.

2.3 데이터 비 암호화 접근제어 기법

데이터 비 암호화 접근제어 기법은 접근제어를 수행하는 시스템을 무한하게 신뢰한다는 가정 하에 시스템이나 사용자가 수립한 정책에 따라 접근인가를 결정하는 방식으로 대부분의 중앙 집중형 SNS에 적용된 기법이다. 하지만 제한적인 정책의 수립만 가능하여 세분화된 접근제어를 지원하지 않으므로 이를 해결하기 위해 사용자 사이에서 맺어지는 관계성을 기반으로 다양한 접근제어 기법들이 제안되었다.

Fong 등은 [12]에서 사용자와 사용자 사이의 관계(user-to-user relationship)를 기반으로 데이터에 대한 접근인가가 결정되는 관계 기반 접근제어 기법을 제안하였다. 이 기법은 이진 관계(binary relation)로 표현된 관계 타입의 배열(relationship sequence)을 정책에 포함시키고 해당 배열을 만족하는 관계를 맺고 있는 사용자만 데이터에 접근을 허가한다. 그리고 두 사용자가 동시에 여러 타입의 관계를 맺을 수 있도록 해주는 다중 관계 타입(multiple relationship type)과 관계의 방향성(directional relationship)을 허용한다. 또한 양상 논리(modal logic)를 적용한 정책 언어를 제공하여 단순한 정책의 조합으로 복잡하고 세분화된 정책 수립을 가능하게 해준다. 하지만 데이터 타깃 행위만을 고려하는 한계점이 존재하고 수신행위에 대한 접근제어만 실시하는 단점이 있다.

Carminati 등은 [13]에서 시맨틱 웹 기술을 활용한 접근제어 기법을 제안하였다. 이 기법은 사용자 사이의 관계뿐만 아니라 사용자와 자원 사이의 관계(user-to-resource relationship)를 정의하여 접근제어에 적용한다. 그리고 인가(authorization), 집행(administration), 필터링(filtering) 이라는 세 가지 유형의 정책을 OWL(Ontology Web Language)와 SWRL(Semantic Web Rule Language)를 사용하여 정의한다. 이 기법은 부분적으로 사용자 타깃 행위에 대한 접근제어를 실시할 수 있고 발신행위에 대한 접근제어도 가능한 장점을 가진다.

Yuan Cheng 등은 [14]에서 정규 표현식(regular expression)을 적용한 사용자 관계 기반 접근제어 기법인 UURAC를 제안하였다. 이 기법은 사용자간의 관계로 구성된 소셜 그래프(social graph) 상에서 관계 경로의 매칭을 통해 접근제어를 수행한다. 그리고 정규 표현식으로 정책을 표현하며 사용자는 경로 규칙의 논리적 조합으로 개별적인 정책을 수립하여 세분화된 접근제어를 수행할 수 있다. 이 기법은 기존에 제안된 기법들처럼 다중 관계 타입과 관계의 방향성을 허용한다. 그리고 사용자 타깃 행위와 데이터 타깃 행위에 대한 접근제어를 모두 수행할 수 있으며 수신행위 뿐만 아니라 발신행위에 대한 접근제어도 가능하여 기존에 제안된 기법들에 비해 더 세분화된 접근제어가 가능한 장점을 가진다.

2.4 기존에 제안된 접근제어 기법의 비교 분석

본 절에서는 기존에 제안된 접근제어 기법들이 앞서 도출한 접근제어 요구사항을 얼마나 만족하는지 분석한다. 아래의 Table 1.과 Table 2.는 각각 데이터 암호화 접근제어 기법과 데이터 비 암호화 접근제어 기법의 요구사항 만족도를 보여준다.

Table 1. Access Control Schemes with Data Encryption

요구사항	데이터 암호화 접근제어 기법		
	Persona	EASiER	Raji
정책 개별화	○	○	○
사용자 타깃	△		
발신 행위			
간접 연결			
동적 접근제어			
데이터 암호화	○	○	○
키 폐기 효율성		○	○

Table 2. Access Control Schemes without Data Encryption

요구사항	데이터 비 암호화 접근제어 기법		
	Fong	Carminati	UURAC
정책 개별화	○	○	○
사용자 타깃		△	○
발신 행위		△	○
간접 연결	○	○	○
동적 접근제어			
데이터 암호화			
키 폐기 효율성			

위의 표와 같이 기존에 제안된 데이터 암호화 접근제어 기법들은 사용자를 타깃 행위에 대한 접근제어를 지원하지 못한다. 그리고 발신행위에 대한 접근제어를 지원하지 않아 악의적인 자료를 게시하는 사용자로의 접근을 차단하기 어렵다. 또한 직접 연결 사용자에 대해서만 그룹을 형성하므로 간접 연결 사용자는 접근이

제한되어 세분화된 접근제어가 불가능하기 때문에 SNS의 특성인 관계의 확장성에 부합되지 않는다. 반면에 데이터 비 암호화 접근제어 기법은 더 세분화된 접근제어를 지원하지만 데이터를 암호화 하지 않으므로 서비스 제공자가 사용자의 모든 데이터에 접근할 수 있는 문제점을 가진다. 그리고 분석된 모든 접근제어 기법들은 사용자 사이의 관계 강도와 강도의 동적인 변화를 고려하지 않아서 동일 그룹 멤버에 대한 차별적 접근제어가 어렵다. 또한 한번 인가된 사용자에게 지속적으로 접근을 허가하는 정적 접근제어만을 제공하는 한계점을 가진다. 이에 본 논문에서는 세분화된 접근제어를 지원하는 UURAC에 사용자 동적 신뢰도 평가 기법을 결합하고 민감한 데이터에 대한 선별적인 암호화 기법을 적용하여 도출된 요구사항을 만족시키는 접근제어 모델을 제안한다.

2.5 UURAC

UURAC는 SNS 사용자를 그래프 상의 정점(node)에, 사용자 사이의 관계는 간선(edge)에 대응시켜 SNS를 그래프의 형태로 표현한다. 그리고 두 사용자 사이를 연결하는 경로(path)를 관계 경로 패턴 (relationship path pattern)로 정의하며 관계 경로 패턴과 사용자 사이의 거리인 홉수(hop count)의 조합을 기반으로 사용자, 시스템 정책을 수립하여 접근제어를 수행한다.

2.5.1 UURAC 구성요소

UURAC는 아래의 Table 3.과 같은 5가지 구성요소로 이루어진다.

Table 3. UURAC Access Control Model Components

구성요소	설명
접근 사용자(u_a)	타깃에 행위를 수행하는 사용자
행위(action)	사용자가 타깃에 수행하는 동작으로 수신행위와 발신행위로 구분
타깃(target)	행위를 수신하는 대상인 타깃 사용자(u_t)와 타깃 자원(r_t)으로 구성되며 시스템은 타깃 자원을 글, 사진, 동영상 등의 유형별로 분류(r.type)하여 식별
접근 요청 (access request)	타깃에 수행하고자 하는 행위에 대한 요청, 〈u_a, action, target 〉
정책(policy)	접근 요청에 대한 허용 여부를 결정짓는 규칙(rule)의 집합으로 사

위의 표와 같이 접근 사용자는 타깃 사용자, 자원에 대한 특정 행위를 수행하기 위해 접근 요청을 한다. 그리고 접근 요청의 허용은 해당 요청과 연관된 주체들이 수립한 정책을 기반으로 SNS 시스템에 의해 결정된다. UURAC에서 정의되는 정책은 아래의 Fig.1.과 같다.

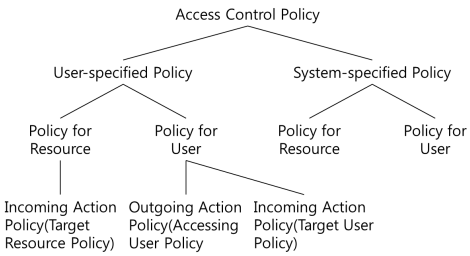

Fig.1. UURAC Access Control Policy Taxonomy

위의 그림과 같이 정책은 정책 수립 주체에 따라 사용자 명시 정책과 시스템 명시 정책으로 구분되며 각 정책은 다시 자원을 위한 정책과 사용자를 위한 정책으로 분류된다. 그리고 사용자는 타깃에 접근을 할 수도 있고 스스로 타깃이 될 수도 있으므로 사용자를 위한 정책은 접근 사용자 정책(AUP), 타깃 사용자 정책(TUP)으로 구분된다. 반면에 자원은 타깃 자원 정책(TRP)만을 가질 수 있고 해당 정책은 자원에 대한 소유권을 가진 사용자인 제어 사용자(u_c)에 의해 수립된다.

2.5.2 소셜 그래프 모델링

UURAC는 SNS를 방향성이 있는 단순 그래프 (directed labeled simple graph)의 형태로 표현

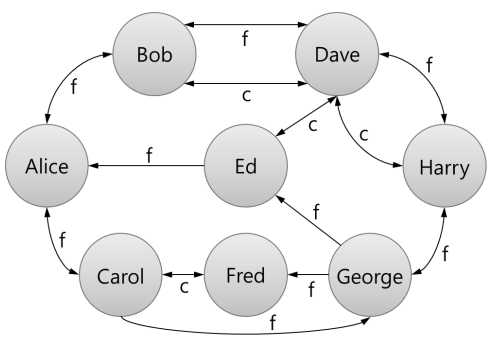

Fig. 2. UURAC Social Graph

한다. 그래프는 트리플 $G=<U,E,\Sigma>$ 로 표기하며 여기서 U는 SNS상의 모든 사용자의 유한집합으로 그래프의 정점이 된다. 그리고 Σ는 사용자 관계 유형 식별자(σ_i)의 유한집합이며 $E \subseteq U \times U \times \Sigma$는 사용자 사이의 관계를 의미하고 그래프의 간선이 된다. 아래의 Fig.2.는 f(friend), c(coworker) 관계 유형으로 구성된 소셜 그래프의 예를 보여준다.

2.5.3 UURAC 정책 명세

UURAC는 관계 경로 패턴으로 구성된 접근제어 정책의 표현을 위해 정규표현식을 적용한다. 그리고 관계 유형이 반복될 경우의 효율적인 표기를 위해 asterisk(*), plus(+), question mark(?)로 구성된 와일드카드 표기법(wildcard notation)을 사용한다. 각 표기는 0 or more, 1 or more, 0 or 1을 의미한다. 접근제어 정책의 구성은 아래의 Fig.3.과 같이 계층적으로 이루어진다.

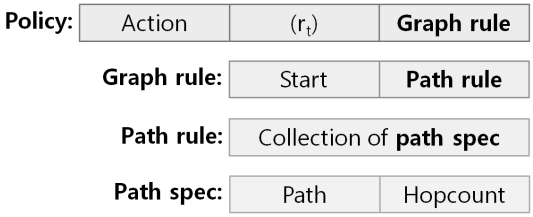

Fig.3. UURAC Policy Specifications

위의 그림과 같이 경로 패턴과 홉수로 구성된 경로 스펙(path spec)의 모음인 경로 규칙(path rule)으로 구성된 그래프 규칙(graph rule)과 타깃에 대한 행위로 정책을 표현한다. 그리고 타깃 자원 정책에는 해당 자원(r_t)도 추가된다. 이러한 구성으로 명세화되는 정책의 종류는 아래의 Table 4.와 같다.

타깃 자원에 대한 접근 요청이 발생하면 시스템은 접근 사용자 정책, 타깃 자원 정책, 자원 시스템 정책을 비교하여 관계 경로 패턴과 거리가 해당 정책들을 만족하는지 분석한다. 예를 들어 Fig.2.에서 Alice가 Harry의 자원인 r_t에 대한 읽기 행위를 요청하면 시스템은 Harry에서 Alice로 연결되는 관계 경로와 거리가 Harry의 타깃 자원 정책 $(read^{-1}, r_t, (f^*cf^*,3))$을 만족하는지 판단한다. 그리고 Alice의 접근 사용자 정책과 시스템의 자원 시스템 정책도 동시에 해석하여 접근 인가를 결정한다.

Table 4. UURAC Access Control Policy Representations

접근 사용자 정책	$<action,(start,path\,rule)>$
타깃 사용자 정책	$<action^{-1},(start,path\,rule)>$
타깃 자원 정책	$<action^{-1},r_t,(start,path\,rule)>$
사용자 시스템 정책	$<action,(start,path\,rule)>$
자원 시스템 정책	$<action,r.type,(start,path\,rule)>$

III. 관계 기반 동적 접근제어 기법 제안

본 논문에서는 기존에 제안된 데이터 비 암호화 접근제어 기법인 UURAC에 사용자의 동적 신뢰도를 측정하여 관계의 강도를 수치화하는 기법[16]을 결합한다. 그리고 프록시 재 암호화 기법 중 하나인 AFGH[22]를 민감한 개인정보의 선별적 암호화에 적용하여 서비스 제공자로부터의 프라이버시 침해를 방지하고 관계의 강도에 따라 동적으로 접근인가를 결정하는 관계 기반 동적 접근제어 모델을 제안한다.

3.1 UURAC와 동적 사용자 신뢰도 평가 기법의 결합

본 논문에서는 UURAC의 접근제어 정책에 사용자 신뢰도 평가 기법을 두 가지의 경우로 나누어 결합한다.

첫 번째는 사용자끼리 처음 관계를 맺는 경우이다. UURAC에서는 관계 경로의 일치성과 사용자 사이의 거리로 간접 연결 사용자에 대한 접근인가를 결정하므로 사용자 사이의 신뢰도가 높다는 가정이 필수적이다. 하지만 페이스북, 트위터 같은 상용화된 SNS에서 사용자의 평균 거리는 5단계정도에 불과하다[23]. 따라서 5이하의 작은 홉수로 정책을 수립해도 접근이 허가될 가능성이 있는 사용자의 신뢰도가 높다고 보장할 수 없다. 따라서 관계를 맺을 때 합리적으로 수치화된 신뢰도 값을 추가하여 높은 신뢰도를 가진 사용자끼리만 관계를 맺을 수 있도록 한다.

두 번째는 접근에 대한 기타 정책을 수립할 경우이다. UURAC의 접근제어에서 한번 접근이 허가된 사용자는 관계의 변경이 발생하지 않는 한 지속적으로 접근이 허가되는 문제점이 있다. 또한 '친구'라는 관계를 맺고 있는 여러 사용자에 대해서 동일한 정책을 수립할 수밖에 없다. 이는 더 친한 친구에게만 보여주고 싶은 사진 등의 자원을 모든 친구에게 보여주게 되는 문제점을 발생시킨다. 따라서 정책을 수립할 때 관계

의 강도를 의미하는 신뢰도 수치를 추가하여 동적이고 더 세분화된 접근제어를 가능하게 한다.

3.1.1 동적 사용자 신뢰도 평가 기법

이창훈 등은 [16]에서 SNS상에서의 사용자 신뢰도를 정의하고 이 정의에 따라 SNS에서의 사용자 총 신뢰도(T)를 공개신뢰도(P)와 관계신뢰도(R)의 조합으로 규정하였다. 여기서 공개신뢰도는 신뢰도를 평가 받는 노드(EN)의 직접 연결 사용자들이 EN에 대해서 평가한 신뢰도의 평균값이며 관계 신뢰도는 한 사용자가 신뢰도 평가 노드(VN)가 되어 다른 사용자에 대해서 일대일로 평가한 신뢰도 값이다. 그리고 각 신뢰도 값은 신뢰도의 정의에 따라 세 가지의 신뢰 파라미터 값인 규범준수정도(N), 가치유사정도(V), 정보교류정도(F)의 조합으로 구성된다. 따라서 규범을 잘 준수하고 비슷한 가치를 공유하며 서로에 대한 교류가 잦은 사용자일수록 높은 신뢰도 값을 갖으며 신뢰도 값은 시스템에 의해 동적으로 측정되어 접근제어 기법에 활용될 수 있다.

3.1.2 신뢰도 평가기법이 적용된 사용자 관계 맺기

SNS에서 사용자 사이에 첫 관계를 맺을 때 적용하기 적합한 신뢰도 값은 공개신뢰도 값이다. 공개 신뢰도 값은 여러 사용자에 의해 측정된 객관적인 값이므로 아직 관계를 맺지 않은 상황에서 친구요청을 하는 사용자가 믿을 만한 사용자인지 판단하는 기준이 될 수 있기 때문이다. 따라서 UURAC의 정책 중 사용자에 대한 정책인 접근 사용자 정책(AUP), 타깃 사용자 정책(TUP), 사용자 시스템 정책(SPU)에 공개신뢰도 최소 요구 값($Pmin$)을 추가한다. 그리고 관계의 용이한 확장을 위해서 경로 규칙(path rule)을 정책에서 삭제하고 대신에 어떤 관계를 맺는지 결정하는 관계 유형 식별자(σ_i)를 추가한다. 또한 관계를 맺기 전에 게시한 데이터에 대한 접근을 방지하는데 활용하기 위해서 새로운 관계를 맺을 때마다 타임스탬프(Tr)를 저장한다. 두 개의 새로운 값이 추가된 정책은 아래의 Table 5.와 같다.

Table 5. Policy Representations for the Rrequest Action

접근 사용자 정책	$< Rrequest, (start, \sigma_i), Pmin >$
타깃 사용자 정책	$< Rrequest^{-1}, (start, \sigma_i), Pmin >$
사용자 시스템 정책	$< Rrequest, (start, \sigma_i), Pmin >$

위의 표와 같이 변경된 정책은 관계 맺기 신청($Rrequest$)이라는 행위에 대한 접근 요청을 할 때만 적용된다. 사용자는 각 σ_i별로 서로 다른 $Pmin$ 값을 설정할 수 있고 관계 맺기를 요청하는 사용자가 믿을 수 있는 사용자인지 관계를 맺기 전에 미리 검증할 수 있으므로 보다 신뢰할 수 있는 접근제어가 가능해진다.

3.1.3 신뢰도 평가기법이 적용된 기타 정책 수립

관계 맺기 요청을 제외한 다른 행위들에 대한 정책을 수립할 때에는 공개신뢰도와 관계신뢰도의 조합인 총 신뢰도를 적용한다. 그리고 직접 연결 사용자와 간접 연결 사용자의 신뢰도 측정 방법에 차이가 있으므로 정책도 두 가지 경우로 구분하여 수립한다.

먼저 직접 연결 사용자에 대한 정책에는 총 신뢰도 최소 요구 값($Tmin$)이 추가된다. 아래의 Table 6.은 $Tmin$이 추가된 정책을 보여주며 Fig.4.는 Alice가 접근 사용자 정책을 $< poke, (u_a, (c, 1)), 60 >$로 수립했을 때 Harry와 Bob에게 poke라는 행위를 수행하는 경우의 접근제어를 보여준다.

Table 6. Policy Representations for the Direct Users

접근 사용자 정책	$< action, (start, path\,rule), Tmin >$
타깃 사용자 정책	$< action^{-1}, (start, path\,rule), Tmin >$
타깃 자원 정책	$< action^{-1}, r_{t,}\,(start, path\,rule), Tmin >$
사용자 시스템 정책	$< action, (start, path\,rule), Tmin >$
자원 시스템 정책	$< action, r.type, (start, path\,rule), Tmin >$

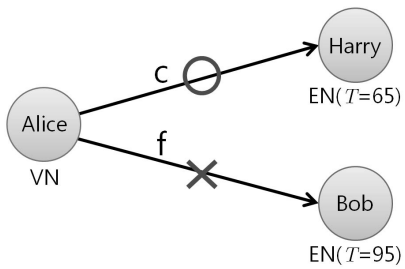

Fig. 4. Access Control for the Direct Users

위의 그림에서 Harry와 Bob 모두 정책에 명시된 최소 요구 값인 60보다 큰 총 신뢰도 값을 가진다. 하지만 Alice와 Bob은 정책에 명시된 관계 경로에 포

함되지 않는 f관계이므로 Bob에 대한 poke 행위는 허용되지 않으며 Harry에 대한 행위만 허용된다.

간접 연결 사용자에 대한 정책 수립을 위해서는 총 신뢰도 값의 계산에 MinMax 방식[24]을 적용한다. MinMax는 두 노드를 연결하는 여러 경로에서 측정된 신뢰도 값들 중 각 경로에서의 최소값을 추려내고 그 중에서 최대값을 선택하는 방식이다. 정책에는 *Tmin* 값 대신에 MinMax로 측정된 총 신뢰도의 최소 요구 값인 *Tminmax*를 추가한다.

아래의 Fig.5.는 Alice가 접근 사용자 정책을 $<read,(u_a,(fc^*,2)),35>$로 수립했을 때 Harry가 소유한 타깃 자원 r_t에 read라는 행위를 수행하는 경우의 접근제어를 보여준다.

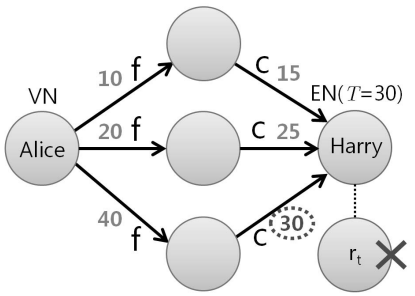

Fig. 5. Access Control for Indirect Users

위의 그림에서 Alice가 측정한 Harry의 총 신뢰도 값은 각 경로에서 측정된 최소값인 10, 20, 30중에서 최대값인 30으로 결정된다. 하지만 정책에 명시된 최소 요구 값인 35보다 작으므로 이 요청은 시스템에 의해 허용되지 않는다.

3.2 사용자 데이터의 선별적 암호화 기법

본 절에서는 사용자가 특히 민감한 개인정보라고 판단한 데이터의 기밀성 보장을 위해서 프록시 재 암호화 기법인 AFGH를 적용한 암호화 기법을 제안한다. 제안하는 암호화 기법에서 데이터는 대칭키 암호 알고리즘으로 암호화 되며 암호화에 사용된 비밀키의 안전한 공유를 위해 AFGH를 사용한다. 시스템은 접근 요청에 대한 접근제어 수행 후에 암호화 된 비밀키의 공유에만 관여할 뿐 비밀키를 직접 복호화 할 수 없으므로 사용자 데이터를 열람할 수 없다.

제안하는 암호화 기법은 타깃 데이터 자원을 소유한 제어 사용자와 해당 데이터에 접근하고자 하는 접근 사용자의 관계에 따라 두 가지 경우로 구분된다.

첫 번째는 제어 사용자와 접근 사용자가 직접 연결된 경우이다. 이 경우에 두 사용자는 이미 특정 관계를 맺고 있으므로 접근 요청이 허가되면 바로 데이터를 복호화 할 수 있도록 비밀키 공유가 수행된다.

두 번째는 제어 사용자와 접근 사용자가 간접 연결된 경우이다. 이 경우에 제어 사용자는 시스템에 의한 접근 요청이 허가되더라도 접근 사용자가 어떤 사용자인지 한차례 더 검증한 후에 비밀키 공유 과정을 수행한다. 이러한 과정을 통해 민감한 데이터에 대한 선별적 암호화와 선별적 복호화 허용이 가능하므로 사용자 프라이버시 보호를 강화할 수 있다.

3.2.1 AFGH 프록시 재 암호화 기법

AFGH는 아래의 조건을 만족하는 두 개의 군(group) G_1, G_2 상에서 수행되는 겹선형 엘가말(bilinear Elgamal) 프록시 재 암호화 기법이다.

1. 군 G_1과 G_2의 위수(order)는 q이다.
2. $g \in G_1$는 군 G_1의 생성자(generator)이다.
3. e는 겹선형 사상(bilinear map) $e: G_1 \times G_1 \rightarrow G_2$ 이다. 즉 모든 $u,v \in G_1$와 $a,b \in \mathbb{Z}_q$에 대해서 $e(u^a,v^b)=e(u,v)^{ab}$를 만족한다.

메시지 송신자와 수신자(복호화 위임자) A, 그리고 복호화 피위임자 B가 글로벌 파라미터 g와 $Z=e(g,g) \in G_2$를 기반으로 수행하는 AFGH 기법은 아래와 같은 과정을 통해 수행된다.

- Key Generation(KG): 사용자 A의 공개키 $pk_a=(Z^{a_1},g^{a_2})$, 개인키 $sk_a=(a_1,a_2)$를 생성함
- Re-encryption Key Generation(RG): 사용자 A는 사용자 B에게 복호화 권한을 위임하기 위해 B의 공개키 g^{b_2}와 자신의 개인키 a_1을 사용해서 재 암호화 키 $rk_{A \rightarrow B}=g^{a_1 b_2} \in G_1$를 생성함
- First-Level Encryption(E_l): 메시지 $m \in G_2$를 pk_a로 암호화 하여 개인키 sk_a를 가지고 있는 사용자 A만 복호화 할 수 있도록 하는 암호문 $c_{a,1}=(Z^{a_1 k},mZ^k)$를 출력함(proxy invisibility를 위해 $c_{a,2}=(Z^{a_2 k},mZ^k)$도 생성할 수 있음)

- Second-Level Encryption(E_2): $m \in G_2$를 pk_a로 암호화하여 A와 복호화 피위임자 B 모두 복호화 할 수 있는 암호문 $c_{a,r} = (g^k, mZ^{a_1 k})$을 출력함
- Re-encryption(R): A가 수신한 second-level 암호문 $c_{a,r} = (g^k, mZ^{a_1 k})$에 $rk_{A \to B} = g^{a_1 b_2}$를 사용하여 B가 복호화 할 수 있도록 재 암호화 함. 즉$c_{a,r}$과 $rk_{A \to B}$로부터 $e(g^k, g^{a_1 b_2}) = Z^{b_2 a_1 k}$를 연산한 후 $c_{b,2} = (Z^{b_2 a_1 k}, mZ^{a_1 k}) = (Z^{b_2 k'}, mZ^{k'})$를 출력함
- Decryption(D_1, D_2): 암호문을 복호화 하는 과정으로 first-level 암호문은 $c_{a,i} = (\alpha, \beta)$에 대해서 $m = \beta / \alpha^{1/a_i} \text{ for } i \in \{1,2\}$를 수행해서 복호화 함. 그리고 second-level 암호문은 $c_a = (\alpha, \beta)$에 대해서 $m = \beta / e(\alpha, g)^{a_1}$을 수행해서 복호화 함.

3.2.2 직접 연결 사용자에 대한 비밀키 공유

직접 연결 사용자에 대한 데이터 암호화는 대칭키 암호인 AES로 수행되며 AES 비밀키를 접근 사용자와 안전하게 공유하기 위해서 AFGH 기법의 second-level encryption을 적용한다.

암호화 과정에 참여하는 개체는 아래와 같으며 Fig.6.은 각 개체의 관계를 보여준다.

- 제어 사용자: 타깃 데이터의 소유자이며 AES와 AFGH 기법으로 데이터와 비밀키를 암호화함
- 접근 사용자: 제어 사용자 소유의 타깃 데이터에 대한 접근을 요청하는 사용자임
- SNS 시스템: 접근제어를 수행하며 AFGH 기법에서 프록시 서버로 동작함
- 제 3 신뢰 기관: 사용자의 공개키에 대한 인증서를 발급하여 사용자의 신원을 보장하는 기관
- SNS 에이전트: 사용자 장치에 설치되어 SNS의 키 관리와 암·복호화 프로세스를 수행하는 프로그램. 소프트웨어 기반 DRM 시스템에서 콘텐츠 암호화키에 대한 임의적 접근을 방지하기 위해 사용되는 소프트웨어 보안영역(secure storage)과 난독화(obfuscation) 기법[25]이 적용되어 암·복호화 연산 중 개인키와 비밀키의 노출을 방지함

위의 그림에서 제 3 신뢰기관은 PKI에서의 인증서

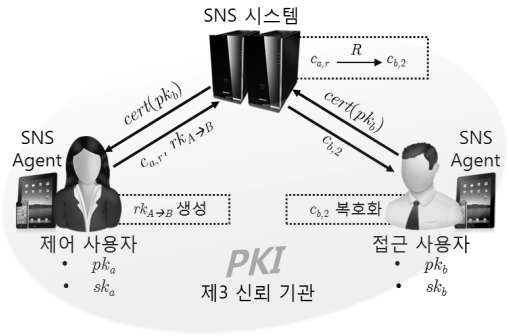

Fig.6. Diagram of the Data Encryption Scheme

발급 기관(certificate authority) 역할을 하여 각 사용자의 공개키에 대한 인증서를 발급한다. 그리고 모든 사용자는 자신의 공개키가 포함된 인증서를 SNS 시스템에 전송하며 SNS 시스템은 인증서 목록을 통해 인증서를 관리하고 다른 사용자들의 요청에 따라 인증서를 전송해준다.

데이터 암호화와 직접 연결 사용자에 대한 비밀키 공유 과정은 구체적으로 아래와 같은 순서로 진행된다. 여기서 제어 사용자는 메시지 송신자, 복호화 위임자의 역할을 하며 SNS 시스템은 프록시 서버로 작동하여 재 암호화를 수행한다. 그리고 접근 사용자는 복호화 피위임자가 되어 재 암호화된 메시지인 AES 비밀키를 복호화 하여 최종적으로 제어 사용자의 데이터를 복호화 한다.

- 제어 사용자는 KG 단계를 통해 공개키 $pk_a = (Z^{a_1}, g^{a_2})$와 개인키 $sk_a = (a_1, a_2)$를 생성하고 제3 신뢰기관으로부터 공개키에 대한 인증서를 발급받아서 SNS 시스템에 전송함
- 제어 사용자는 다른 사용자들과 관계를 형성할 때 해당 사용자들의 인증서를 SNS 시스템으로부터 전송받아 검증함. 그리고 각 사용자에 대한 rk를 생성하고 모든 rk를 SNS 시스템에 전송함
- 제어 사용자는 $m \in G_2$를 랜덤하게 선택하여 AES 비밀키를 생성하고 민감한 데이터를 선별하여 AES로 암호화함. 그리고 E_2 단계로 비밀키를 암호화 하여 $c_{a,r} = (g^k, mZ^{a_1 k})$를 출력하고 이 값과 암호화된 데이터, 데이터 생성 타임스탬프(Td)를 SNS 시스템에 전송함
- 접근 사용자는 제어 사용자의 게시판에서 암호화된 데이터의 요약 정보를 확인한 후 SNS 시스템

에게 접근 요청을 함

- SNS 시스템은 접근 사용자 정책, 제어 사용자 정책, 자원 시스템 정책을 해석하고 타임스탬프 Tr과 Td를 비교하여 접근 요청에 대한 허가를 결정함. 만약 접근 요청이 허가되면 SNS 시스템은 제어 사용자로부터 미리 전송받은 재 암호화 키 $rk_{A\to B}$를 사용해서 $c_{a,r}$를 재 암호화 하여 $c_{b,2}=(Z^{b_2a_1k},mZ^{a_1k})=(Z^{b_2k'},mZ^{k'})$를 출력함. 그리고 이 값과 AES로 암호화 된 제어 사용자의 데이터를 접근 사용자에게 전송함
- 접근 사용자는 $c_{b,2}$를 자신의 개인키 b_2로 복호화 하여 AES 비밀키 $m\in G_2$을 얻어내고 암호화된 제어 사용자의 데이터를 복호화 하여 평문 데이터를 획득함

위와 같은 과정을 통해 제어 사용자와 직접적인 관계를 맺고 있는 접근 사용자는 접근 요청의 허가와 함께 곧바로 타깃 데이터를 전송받아 열람할 수 있다. 아래의 Fig.7.은 이러한 과정의 프로토콜을 나타낸다.

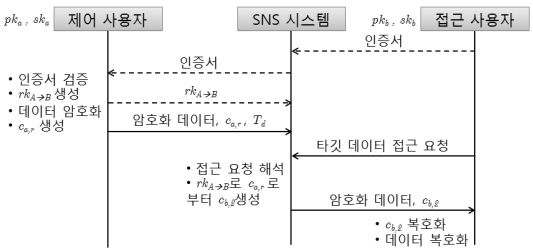

Fig.7. Protocol for Direct Friends

3.2.3 간접 연결 사용자에 대한 비밀키 공유

간접 연결 사용자에 대한 비밀키 공유 과정은 SNS 시스템으로부터 타깃 데이터에 대한 접근 요청이 허가되어도 곧바로 수행되지 않는다. 간접 연결 사용자를 위한 재 암호화 키는 접근 요청 이전에 미리 생성할 수 없기 때문이다. 대신에 제어 사용자는 접근 사용자로부터의 데이터 접근요청을 확인하면 인증서를 통해 해당 사용자를 검증하여 신뢰할 수 있는 사용자라고 판단이 된 경우에만 재 암호화 키 $rk_{A\to B}$를 생성해서 나머지 과정을 수행한다.

데이터 암호화와 간접 연결 사용자에 대한 비밀키 공유 과정은 아래와 같은 순서로 진행되며 제어 사용자, SNS 시스템, 접근 사용자의 역할은 직접 연결 사용자에 대한 비밀키 공유 과정과 동일하다.

- 제어 사용자는 $m\in G_2$를 랜덤하게 선택하여 AES 비밀키를 생성하고 민감한 데이터를 선별하여 AES로 암호화함. 그리고 E_2 단계로 비밀키를 암호화 하여 $c_{a,r}=(g^k,mZ^{a_1k})$을 출력하고 암호화 된 데이터와 함께 SNS 시스템에 전송함
- 접근 사용자는 제어 사용자의 게시판에서 암호화 된 데이터의 요약 정보를 확인한 후 SNS 시스템에게 접근 요청을 함
- SNS 시스템은 접근 사용자 정책, 제어 사용자 정책, 자원 시스템 정책을 해석하여 접근 요청에 대한 허가를 결정함. 만약 접근 요청이 허가되면 SNS 시스템은 해당 요청이 허가되었다는 사실을 제어 사용자에게 통보하고 접근 사용자의 인증서를 제어 사용자에게 전송함
- 제어 사용자는 통보를 확인하고 접근 사용자의 인증서를 검증하여 해당 사용자에 대한 타깃 데이터 열람 허가 여부를 결정함. 만약 열람을 허가할 경우 인증서와 자신의 개인키로부터 $rk_{A\to B}$를 생성하여 SNS 시스템에 전송함
- SNS 시스템은 제어 사용자로부터 받은 $rk_{A\to B}$를 사용해서 $c_{a,r}$를 재 암호화 하여 $c_{b,2}=(Z^{b_2a_1k},mZ^{a_1k})=(Z^{b_2k'},mZ^{k'})$를 출력함. 그리고 이 값과 AES로 암호화 된 제어 사용자의 데이터를 접근 사용자에게 전송함
- 접근 사용자는 $c_{b,2}$를 자신의 개인키 b_2로 복호화 하여 AES 비밀키 $m\in G_2$을 얻어내고 암호화된 제어 사용자의 데이터를 복호화 하여 평문 데이터를 획득함

위와 같은 과정을 통해 제어 사용자와 간접 연결된

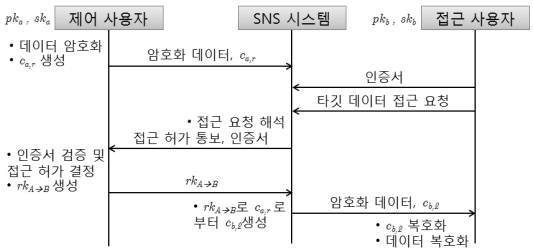

Fig.8. Protocol for Indirect Friends

접근 사용자는 SNS 시스템으로부터의 접근 요청 허가와 제어 사용자로부터의 비밀키 공유 허가를 모두 받아야 타깃 데이터를 전송받아 열람할 수 있다. 따라서 제어 사용자는 어떤 사용자에게 타깃 데이터의 열람을 허락할 지 선별적으로 선택할 수 있으므로 간접 연결 사용자에 대한 더 세분화된 접근제어가 가능하다. 아래의 Fig.8.은 이러한 과정의 프로토콜을 나타낸다.

IV. 제안하는 접근제어 기법 분석

4.1 제안 기법의 접근제어 특성 분석

본 논문에서 제안하는 기법은 기존에 제안된 UURAC의 접근제어 특성을 그대로 따른다. 따라서 접근제어 정책의 개별화, 사용자 타깃 행위에 대한 접근제어, 발신행위에 대한 정책수립, 간접 연결 사용자에 대한 접근제어를 모두 지원한다. 그리고 제안하는 기법은 사용자 동적 신뢰도의 최소 요구 값을 접근제어정책에 추가하여 신뢰도 값의 변화에 따른 동적 접근제어를 지원한다. 특히 사용자 사이에서 처음 관계를 형성할 때 객관적 신뢰도 수치인 공개신뢰도 값을 통해서 악의적인 사용자와의 관계 맺음을 방지할 수 있으므로 관계 경로의 일치성을 기준으로 수행되는 관계 기반 접근제어의 신뢰성이 향상된다.

4.2 제안 기법의 데이터 암호화 특성 분석

제안 기법에서 사용자는 SNS 시스템으로부터 자신의 프라이버시를 보호하기 위해서 AES 암호화 알고리즘으로 데이터를 암호화 한다. 그리고 AES 비밀키를 접근 사용자와 공유하기 위해서 프록시 재 암호화 기법인 AFGH 기법을 적용한다. 여기서 SNS 시스템은 프록시 서버로 동작하여 제어 사용자의 공개키로 암호화 된 AES 비밀키의 재 암호화에만 관여한다. 그리고 재 암호화 키 $rk_{A\rightarrow B}$는 제어 사용자의 개인키와 접근 사용자의 공개키로 구성되므로 SNS 시스템이 자의적으로 생성할 수 없다. 따라서 제어 사용자가 접근을 허가하고자 하는 접근 사용자만 AES 비밀키를 공유할 수 있으며 SNS 시스템은 암호화된 제어 사용자의 데이터를 열람할 수 없으므로 제어 사용자의 프라이버시가 보호된다.

제안 기법에 대해서 SNS 시스템이 시도할 수 있는 공격에는 접근 사용자로의 위장 공격, 접근 사용자와의 공모 공격이 있다. 하지만 접근 사용자의 공개키는 제 3의 기관에 의해 서명된 인증서에 포함되므로 제어 사용자는 인증서의 검증을 통해 접근 사용자의 신원을 확인할 수 있다. 따라서 SNS 시스템은 접근 사용자로 위장하기 어렵다. 그리고 SNS 시스템이 접근 사용자와 공모하여도 접근 사용자의 개인키는 SNS 에이전트에 의해 임의적인 유출이 차단되므로 $rk_{A\rightarrow B}$를 통해 $rk_{A\rightarrow C}$나 $rk_{A\rightarrow D}$ 등을 유도해낼 수 없다. 또한 접근 사용자가 복호화한 제어 사용자의 AES 비밀키도 SNS 에이전트에 의해 유출이 방지되므로 접근 사용자는 이 비밀키를 마음대로 열람하거나 SNS 시스템에게 전송할 수 없다.

제안 기법은 사용자 연산 효율성 측면에서 장점을 가진다. 데이터는 속도가 빠른 AES로 암호화 되고 최대 256bit 길이의 AES 비밀키만 AFGH 기법으로 암호화 되므로 연산 부하를 줄일 수 있기 때문이다. 그리고 AFGH 기법에서의 연산 부하는 타원곡선 상의 겹선형 사상에 사용되는 페어링 연산(pairing computation)에서 가장 크게 발생하는데 제안 기법에서는 SNS 시스템이 수행하는 재 암호화 과정에서만 페어링 연산이 수행되며 사용자가 수행하는 암호화, 복호화 과정에서는 일반적인 타원곡선 상의 점에 대한 연산만 수행된다. 따라서 사용자 측의 연산 부하가 상대적으로 낮다. 구체적인 연산 수행 시간은 타원곡선과 페어링의 선택에 따라 결정되며 페어링 연산의 효율성 향상을 위해 Barreto-Naehrig(BN) curve, tate pairing, ate pairing, optimal ate pairing과 같은 다양한 타원곡선, 페어링의 적용이 가능하다[26][27].

아래의 Table 7.은 일반 PC와 모바일 기기의 하드웨어 환경에서 타원곡선과 페어링의 선택에 따른 사용자, SNS 시스템에서의 AFGH 암·복호화, 재 암호화(페어링 연산) 수행 시간을 나타낸다.

Table 7. Average Operation Times of the AFGH, Pairing

구분	파라미터	암호화	재 암호화 (페어링 연산)	복호화
[22]	256bit	3.3ms	8.7ms	1.5ms
[26]		-	54.19ms	-
[27]	224bit	-	135ms	-

위의 표에서 [22]는 Intel Pentium 4 2.8GHz,

1GB RAM, Debian Linux 2.6.8 환경에서 supersingular 타원곡선 상의 tate pairing을 사용하여 AFGH 기법을 수행했을 때의 연산 시간을 보여준다. 그리고 [26]은 ARM Nvidia Tegra 2(ARM Coretex A9) dual core 1GHz, 1GB DDR2 RAM 환경에서 Barreto-Naehrig(BN) curve 상의 optimal ate pairing을 사용하여 페어링 연산을 수행했을 때의 연산 시간을 나타내며, [27]은 Apple A4(ARM Coretex A8), 512MB DDR RAM, iOS 4 환경에서 MNT 타원곡선 상의 페어링 연산을 수행했을 때의 연산 시간을 나타낸다. 여기서 [22]의 수행시간 값을 통해 페어링 연산을 포함하지 않는 암·복호화 과정의 연산 시간이 재 암호화 과정보다 적게 걸리므로 사용자 측면에서의 연산 부하가 SNS 시스템보다 낮다는 것을 확인할 수 있다. 그리고 모바일 기기의 프로세서에서 측정된 [26], [27]의 페어링 수행시간 값을 통해서 모바일 기기에서의 AFGH 기법 암·복호화 연산 수행이 현실적으로 가능하다는 것을 예측할 수 있다.

제안 기법은 키 폐기 효율성에서도 장점을 가진다. 기존의 데이터 암호화 접근제어 기법들은 새로운 그룹 멤버의 추가나 기존 멤버의 삭제에 따른 키 전방향 안전성(forward secrecy)과 후방향 안전성(backward secrecy)을 위해 키 폐기 및 재분배 과정이 필요하다. 하지만 제안 기법에서는 타임스탬프 Tr, Td의 비교를 통해 관계가 형성되기 이전의 데이터에 대한 접근을 방지할 수 있으며 관계가 종료 될 경우 시스템이 수행하는 접근제어에 의해 데이터에 대한 접근이 차단되므로 부가적인 키 폐기 및 재분배를 수행할 필요가 없다. 아래의 Table 8.은 이와 같은 특성들을 기반으로 분석한 제안 기법의 접근제어 요구사항 만족 여부를 보여준다.

Table 8. Analysis of the Proposed Access Control Scheme

요구사항	제안 기법의 요구사항 만족도
정책 개별화	○
사용자 타깃	○
발신 행위	○
간접 연결	○
동적 접근제어	○
데이터 암호화	○
키 폐기 효율성	○

V. 결 론

소셜 네트워크 서비스는 온라인상에서 정보의 공개성과 관계의 확장성을 기반으로 사용자의 개성을 표현하고 인적 네트워크를 강화시켜주는 서비스이다. 하지만 SNS의 이러한 특성은 사용자의 민감한 개인정보를 무분별하게 노출시키고 사용자로 하여금 신뢰할 수 없는 정보를 열람하도록 유도하여 다양한 부작용을 초래한다. 특히 이러한 부작용 중 가장 문제가 되는 것은 사용자의 프라이버시 침해 문제이다. 이에 본 논문에서는 SNS에서의 사용자 프라이버시 보호를 위한 접근제어 요구사항을 도출하고 기존에 제안된 접근제어 기법들을 데이터 암호화 여부를 기준으로 구분하여 비교 분석하였다. 그리고 기존에 제안된 기법들의 한계점을 개선하기 위해서 데이터 비 암호화 접근제어 기법인 UURAC에 사용자 동적 신뢰도 측정 기법을 결합하였다. 또한 사용자의 민감한 데이터에 대한 선별적 암호화, 복호화를 위해 AES와 AFGH 프록시 재 암호화 기법을 적용하였다. 하지만 사용자 신뢰도의 통계적 분석을 통해 합리적인 신뢰도 최소 요구 값의 범위를 정하기 위한 추가적인 연구가 필요하고 SNS 시스템에 저장되는 재 암호화 키의 저장 공간 효율성을 최적화하기 위한 방법의 적용이 요구된다.

본 논문에서 제안된 기법은 자신의 개성을 표현하고 사회적 관계를 유지, 확장하고자 하는 SNS 사용자의 요구사항에 부합하며 중앙 집중방식으로 제공되는 상용 SNS 서비스에 적용 가능하다. 그리고 기존의 접근제어 기법들과 비교해서 더 세밀하고 동적인 접근제어를 지원하고 SNS 시스템으로부터의 프라이버시 침해를 방지할 수 있으며 데이터의 암·복호화에 따른 사용자 측면의 연산과 키 폐기 효율성에서 장점이 있다. 따라서 개인정보의 무분별한 수집과 악용을 방지하는 안전한 SNS 시스템 구축 및 사용 환경의 구성에 기여할 수 있을 것이다.

References

[1] NB Ellison, "Social network sites: definition, history, and scholarship," Journal of Computer Mediated Communication, vol. 13, no. 1, pp. 210-230, Oct. 2007.

[2] KY Lin and HP Lu, "Why people use social networking sites: an empirical study integrating network externalities and mo-

tivation theory," Computers in Human Behavior, vol. 27, no. 3, pp. 1152-1161, May. 2011.
[3] R Gross and A Acquisti, "Information revelation and privacy in online social networks," ACM workshop on Privacy in the electronic society, pp. 71-80, Nov. 2005.
[4] B Debatin, et al. "Facebook and online privacy: attitudes, behaviors, and unintended consequences," Journal of Computer Mediated Communication, vol. 15, no. 1, pp. 83-108, Oct. 2009.
[5] S Ye and S. F Wu, "Measuring message propagation and social influence on Twitter. com," Springer Social informatics, LNCS 6430, pp. 216-231, Oct. 2010.
[6] B Krishnamurthy and CE. Wills, "On the leakage of personally identifiable information via online social networks," 2nd ACM workshop on Online social networks, pp. 7-12, Aug. 2009.
[7] G Hogben, "Security issues and recommendations for online social networks," ENISA Position Paper, no. 1, Oct. 2007.
[8] C Zhang, et al. "Privacy and security for online social networks: challenges and opportunities," IEEE Network, vol. 24, no. 4, pp. 13-18, Aug. 2010.
[9] R Baden, et al. "Persona: an online social network with user-defined privacy," ACM SIGCOMM Computer Communication Review, vol. 39, no. 4, pp. 135-146, Aug. 2009.
[10] S Jahid, P Mittal, and N Borisov, "EASiER: Encryption-based access control in social networks with efficient revocation," 6th ACM Symposium on Information, Computer and Communications Security, pp. 411-415, Mar. 2011.
[11] F Raji, et al. "Online social network with flexible and dynamic privacy policies," CSI International Symposium on. IEEE, pp. 135-142, Jun. 2011.
[12] PW.L. Fong and I Siahaan, "Relationship-based access control policies and their policy languages," 6th ACM symposium on Access control models and technologies, pp. 51-60, Jun. 2011.
[13] B Carminati, et al. "Semantic web-based social network access control," computers & security, vol. 30, no. 2, pp. 108-115, Aug. 2010.
[14] Y Cheng, J Park, and R Sandhu, "A user-to-user relationship-based access control model for online social networks," Data and Applications Security and Privacy XXVI, LNCS 7371, pp. 8-24, Jul. 2012.
[15] L Banks, SF Wu, "All friends are not created equal: an interaction intensity based approach to privacy in online social networks," CSE 09. International Conference on IEEE, vol. 4, pp. 970-974, Aug. 2009.
[16] CH Lee, et al. "Dynamic user reliability evaluation scheme for social network service," Journal of the Korea Institute of Information Security and Cryptology, 23(2), pp. 157-168, Apr. 2013.
[17] J Park, R Sandhu, and Y Cheng, "Acon: activity-centric access control for social computing," Availability, Reliability and Security (ARES), 2011 Sixth International Conference on IEEE, pp. 242-247, Aug. 2011.
[18] J Park, R Sandhu, and Y Cheng, "A user-activity-centric framework for access control in online social networks," Internet Computing, vol. 15, no. 5, pp. 62-65, Oct. 2011.
[19] S Guha, K Tang, and P Francis, "NOYB: Privacy in online social networks," 1st workshop on Online social networks ACM, pp. 49-54, Aug. 2008.
[20] MM. Lucas and N Borisov, "Flybynight: mitigating the privacy risks of social networking," 7th ACM workshop on Privacy

in the electronic society, pp. 1-8, Oct. 2008.
[21] A Shamir, "How to share a secret," Communications of the ACM, vol. 22, no. 11, pp. 612-613, Nov. 1979.
[22] G Ateniese, et al. "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Transactions on Information and System Security, vol. 9, no. 1, pp. 1-30, Feb. 2006.
[23] L Backstrom, et al. "Four degrees of separation," 3rd Annual ACM Web Science Conference, pp. 33-42, Jun. 2012.
[24] HS Song, "A study on transitivity and composability of trust in social network," Journal of Information Technology Applications & Management, 18(4), pp. 41-53, Oct. 2011.
[25] J Nützel, and A Beyer, "Towards trust in digital rights management systems," Trust and Privacy in Digital Business, LNCS 4083, pp. 162-171, Sep. 2006.
[26] T Acar, et al. "Affine pairings on ARM," Pairing - Based Cryptography, Pairing 2012 5th International Conference, LNCS 7708, pp. 203-209, May. 2012.
[27] J. A. Akinyele, et al. "Securing electronic medical records using attribute-based encryption on mobile devices," 1st ACM workshop on security and privacy in smartphones and mobile devices, pp. 75-86, Oct. 2011.

〈저자소개〉

권 근 (Keun Kwon) 학생회원
2011년 8월: 성균관대학교 컴퓨터공학과 졸업
2011년 8월~현재: 성균관대학교 전자전기컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 암호이론, 네트워크 보안

정 영 만 (Youngman Jung) 학생회원
2012년 2월: 성균관대학교 수학과 졸업
2012년 3월~현재: 성균관대학교 전자전기컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 암호이론

정 재 욱 (Jaewook Jung) 학생회원
2010년 2월: 한국항공대학교 전자전기컴퓨터공학과 졸업
2012년 2월: 성균관대학교 전자전기컴퓨터공학과 석사
2012년 3월~현재: 성균관대학교 전자전기컴퓨터공학과 박사과정
〈관심분야〉 정보보호, 암호학, 네트워크보안, 포렌식

최 윤 성 (Younsung Choi) 학생회원
2011년 8월: 성균관대학교 컴퓨터공학과 졸업
2011년 8월~현재: 성균관대학교 전자전기컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 전자공학, 통신공학

전 웅 렬 (Woongryul Jeon) 학생회원
2011년 8월: 성균관대학교 컴퓨터공학과 졸업
2011년 8월~현재: 성균관대학교 전자전기컴퓨터공학과 석사과정
〈관심분야〉 정보보호, 전자공학, 통신공학

원 동 호 (Dongho Won) 종신회원
1976년~1988년: 성균관대학교 전자공학과(공학사, 공학석사, 공학박사)
1978년~1980년: 한국전자통신연구원 전임연구원
1985년~1986년: 일본 동경공업대 객원연구원
1988년~2003년: 성균관대학교 교학처장, 전기전자 및 컴퓨터공학부장, 정보통신대학원장, 정보통신기술연구소장, 연구처장
1996년~1998년: 국무총리실 정보화추진위원회 자문위원
2002년~2003년: 한국정보보호학회장
현재: 성균관대학교 전자전기컴퓨터공학과 교수, 한국정보보호학회 명예회장
〈관심분야〉 정보보호 암호이론, 정보이론