情報保護學會論文誌
第 21 卷 第 1 號, 2011. 2

소셜 네트워크 서비스를 위한 프라이버시 보호 정책언어 및 프라이버시 보호 모듈 구현*

김 지 혜,[1†] 이 형 효,[2‡]
[1]전남대학교, [2]원광대학교(정보과학연구소)

Implementation of Privacy Protection Policy Language and Module For Social Network Services*

Ji-hye Kim,[1†] HyungHyo Lee,[2‡]
[1]Chonnam National University, [2]Wonkwang University

요 약

소셜 네트워크 서비스는 현실 세계의 인간관계를 바탕으로 온라인에서 서비스를 제공하는 웹서비스로 서비스의 인기가 높아짐에 따라 프라이버시 보호, 즉 개인정보 소유자의 권리 보호에 대한 목소리가 높아졌다. 본 논문은 개인정보 소유자의 자기정보 통제권을 보장하면서 소셜 네트워크 간 데이터 공유를 지원하기 위한 정책언어를 제시하고 이 정책을 기반으로 하는 프라이버시 보호 모듈을 설계 및 구현한다. 프라이버시 보호를 위한 정책언어는 개인정보에 접근하는 사용자의 속성을 기반으로 접근여부를 결정하는 속성기반 접근통제모델을 바탕으로 하고 있다. 뿐만 아니라 본 논문에서 제시된 정책 언어와 개발 모듈은 소셜 네트워크 서비스 간 안전한 데이터 공유 외에도 개인정보에 대한 자기정보 통제권이 필요한 다른 응용분야에 적용이 가능하다.

ABSTRACT

An SNS(Social Network Service) enables people to form a social network on online as in the real world. With the rising popularity of the service, side effects of SNSs were issued. Therefore we propose and implement a policy-based privacy protection module and access control policy language for ensuring the right of control of personal information and sharing data among SNSs. The policy language for protecting privacy is based on an attribute-based access control model which grants an access to personal information based on a user's attributes. The policy language and the privacy protection module proposed to give the right of control of personal information to the owner, they can be adopted to other application domains in which privacy protection is needed as well as secure sharing data among SNSs.

Keywords: Privacy Protection, SNS, Attribute-based access control policy

I. 서 론

소셜 네트워크 서비스는 이용자들의 관계에 기반한 서비스들을 말하며 웹 서비스 업계에서 큰 주목을 받았다. 대표적인 소셜 네트워크 서비스로는 MySpace, Facebook, Linkdein 등이 있고, 국내에는 Cyworld가 대표 서비스라고 할 수 있다. 특히 이런 서비스들은 사용자의 아이덴티티 정보를 기반으로 서비스를 제공하기 때문에 아이덴티티 관리측면과 프라이버시보호 측면에서 연구할 가치가 있다.

기존의 소셜 네트워크 서비스 사이트들은 다음과

접수일(2010년 3월 18일), 수정일(2010년 8월 14일), 게재확정일(2010년 9월 14일)
* 이 논문은 2009년도 원광대학교의 교비지원에 의해서 수행됨.
† 주저자, jihekim@kisa.or.kr
‡ 교신저자, hlee@wonkwang.ac.kr

같은 문제점을 가지고 있다[1]. 첫째, 사이트들이 단독적인 정보 저장소를 형성하고 있으며, 정보들은 대부분의 경우 해당 서비스 사이트를 제외한 다른 서비스 사이트들에서 사용할 수 없다. 둘째, 자기 개인정보에 대한 완전한 통제권을 사용자에게 허가하지 않아 잠재적인 프라이버시 문제를 가지고 있다. 물론 각 서비스 업체들은 APIs와 플랫폼을 서비스 차원에서 혹은 흩어져있는 개인 데이터의 공유 목적으로 공개하고 있다. 하지만 각 메이저 사업자들이 공개한 APIs나 플랫폼을 기반으로 만들어진 어플리케이션들은 단지 부분적 데이터 공유만을 허락하며, 이마저도 아직까지 이렇다 할 표준이 없다. 이런 문제를 극복하기 위해 특정 서비스 제공 업체가 아닌 개인이 운영하는 온톨로지 기반의 소셜 네트워크 서비스를 제안하는 아이디어도 거론 되고 있다.

이런 문제를 해결하기 위해 소셜 네트워크 서비스 사이트는 개인이 원하는 정보를 원하는 사람과 공유하고 원하지 않는 접근에 대해 제어가 가능 하도록 통제권을 정보 소유자에게 보장해 주어야 한다. 개인이 데이터를 생성 할 때 사용자가 접근통제정책을 설정하고 적용하는 것이 필요하며, 이 정책은 사용자의 프라이버시를 최대한 보장하면서 보다 안전한 방법으로 데이터를 공유할 수 있는 방법을 제공해야한다. 본 논문은 데이터에 접근하는 대상의 접근 권한을 개인정보 소유자가 정한 정책을 기술 할 수 있는 접근통제 정책언어를 제안하고, 이를 평가해서 접근 통제를 실행하는데 필요한 정책 결정 모듈을 설계 및 구현한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 소셜 네트워크 서비스에서 개인정보에 대한 접근 통제를 실현하기 위해 다양한 정책언어와 접근통제 모델을 살펴본다. III장에서는 현재 소셜 네트워크 서비스의 문제점과 요구사항을 기반으로 제안하고자 하는 프라이버시가 강화된 접근 통제 모델 언어를 제시한다. IV장은 접근 통제 모델 언어를 이용한 정책 기반의 접근 통제 모듈을 설계 및 구현 하며 이를 간단히 구성한 소셜 네트워크 서비스사이트에 연동하여 시험하여 본다. V장에서는 결론 및 향후 연구방향에 대해 기술한다.

II. 관련 연구

사용자들이 누가 자신의 개인 데이터에 접근 가능한지 구분할 수 있는 프라이버시 메뉴를 가지고 있더라도 모든 개인정보가 개인정보 주체의 의지데로 관리되지는 않는다[2]. 예를 들어, MySpace에서 어떤 사용자의 동의 없이 50만이 넘는 이미지들이 세어 나갔다는 보고가 있다[3][4].

이를 해결하기 위해서 소셜 네트워크 서비스 환경에서 개인정보와 프라이버시를 보호하기 위한 접근통제를 제공하는 시스템이 필요하다. 이런 접근통제를 수행하는 시스템 구성요소는 시스템에서 외부로부터 오는 개인정보에 대한 접근 요청을 사용자가 명시한 사용자 정책과 비교하여 인가결정을 내리는 역할을 한다. 이를 통해서 데이터를 요구하는 주체가 해당 데이터에 연산을 가할 권한이 있는 가를 판단하여 권한이 있는 사용자에게만 데이터에 대한 접근을 허가 할 수 있는 것이다. 이와 같은 시스템 상의 구성 요소가 개인정보 소유자의 프라이버시를 보호하기 위해 가장 좋은 방법은 개인정보 소유자의 정책을 기반으로 요구의 타당성을 판단하는 것이다. 따라서 사용자의 정책을 표현하는 방법이 필요하다. 정책을 표현하는 방법은 여러 방법이 있겠으나 접근통제 정책 언어를 사용하는 것이 가장 좋은 방법이 될 것이다. 이런 접근통제 정책을 기술 할 수 있는 언어는 XML을 기반으로 하는 것이 가장 알맞다. 그 이유는 XML은 플랫폼과 시스템에 독립적이고, 이를 이용할 경우 머신과 사람이 동시에 이해 가능하다. 또 한 XML은 문법과 의미를 확장 가능하여 보안 정책을 작성하는데 필요한 요구사항들을 표현하는데 유연성을 제공하기 때문이다.

2.1 P3P

P3P(Platform for Privacy Preferences) 프로젝트는 1997년부터 W3C 주도하에 온라인 개인정보보호 문제를 해결하기 위해 개발이 시작되어, 사용자 에이전트가 웹 사이트의 개인정보보호 정책을 자동적으로 검색하여 쉽게 해석할 수 있도록 개인정보보호 정책의 표준화된 형식을 개발하였다[5, 6].

P3P는 개인정보보호 정책을 표현하고 정책에 의해 개인정보 제공여부를 판단하는 것을 보장하나 그 목적자체가 웹 사이트의 정보 수집 관련 정책을 알려주기 위해 설계된 것으로 정보 수집 및 정보이용 정책을 정보를 수집 할 때 알려주는 것이지 수집 이후의 문제를 다루지 않는다. 따라서 소셜 네트워크 서비스에서 사용자 정책을 통해 정보의 제공을 통제하기 위해서는 목적자체가 맞지 않다. 소셜 네트워크 서비스의 사용자 정책은 매우 개인적이고 각 개인정보 소유자 별로 각각 작성 및 적용 가능해야 한다. 그러나 P3P의 정책은 사이트의 개인정보보호정책을 기술하기 위한

XML형식을 제공하므로 이는 사용자의 정책을 나타내기에 알맞지 않다. 또 정책을 명시하는 방법은 인간의 언어로 되어 있는 개인정보보호정책을 P3P문법으로 변환한 다음, 그 결과물로부터 생성된 파일들을 고지하는 동시에 전체 사이트 중 정책이 적용되는 부분을 알려주는 참조 정책 파일 작성 등 불필요한 절차들이 있어 복잡하고 불편하다.

2.2 EPAL

EPAL(Enterprise Privacy Authorization Language)은 IT 시스템안의 데이터를 제어 및 통제하기 위해 접근 권한 허가에 관한 기업차원의 프라이버시 정책을 작성하기 위한 형식의 언어이다[7]. EPAL은 기업환경에서 정책을 바탕으로 기업이 고객의 정보를 이용하는데 있어 프라이버시를 보호할 수 있는 방안과 정책 기반의 접근통제를 실현하고 있다. 따라서 기업 간 개인정보 공유를 위해서 기업이 서로 어휘를 알고 있을 때에만 정책을 번역할 수 있다는 단점이 있다. 또 사용자라는 정책을 적용 받는 주체가 미리 정의 되어 분류되어 있고, 만들어진 목적 자체가 기업에서 고객의 정보를 이용하기 위함이므로 주체가 주로 기업 내 역할에 기반하고 있기 때문에 소셜 네트워크 서비스 환경에서 개인정보에 대한 접근을 허용하기 위해 주체를 식별하기 위한 방법으로 잘 어울리지 않는다.

2.3 XACML

접근제어 시스템은 요청자로부터 데이터에 대한 접근 질의를 받으면 그 요청에 따른 보안정책을 적용시켜 접근에 관한 판단을 하고 허용여부를 결정 한다. 이런 과정에 필요한 요청자의 질의를 정형적인 형식으로 표현하는 컨텍스트 스키마와 보안정책을 표현하는 정책 스키마를 제공하는 OASIS의 표준 정책 언어가 XACML(Extensible Access Control Markup Language)이다[8]. XACML은 정보시스템 보안정책을 표현하기 위해 필요한 요구사항에 대한 해결방법을 제공한다. 먼저 규칙과 정책 결합에 대해서 보안정책을 표현하기 위해서는 요청자의 자원 요청에 적용할 수 있는 규칙이나 정책들을 세 개의 최상위수준 정책 요소를 정의하여 단일 정책집합으로 결합하는 방법을 제공한다. 두 번째로 결합 알고리즘이다. 적용할 규칙이나 정책들이 여러 개 존재할 때 이들을 평가한 결과를 결합하는 절차를 정의할 수 있도록 XACML은 RuleCombiningAlgID나 PolicyCombiningAlgID 속성으로 식별되는 결합 알고리즘을 정의하고 있다. 세 번째는 다중주체이다. 접근통제 정책들은 종종 하나 이상의 주체(subject)를 인식해야 하는 행위들을 다루기도 해야 하기 때문에, XACML은 자원요청과 관련된 요청 컨텍스트에 하나 이상의 주체를 포함시킬 수 있도록 한다. 네 번째는 주체와 자원 속성들에 기반을 둔 정책 명세이다. 권한부여 결정을 내리는데 있어서 요구사항은 주체의 신원보다는 일부 속성에 기초하여 인가결정을 할 수 있도록 한다. 다섯 번째는 다중 값을 지닌 속성들 이다. LDAP이나 XPath, SAML 등 대부분의 기술들은 속성을 주고받는데 있어서 속성이 다중 값을 지니도록 허용한다. 여섯 번째는 자원 내용에 기반을 둔 정책 명세이다. 많은 응용들 중에는, 접근하는 정보자원에 포함되어 있는 데이터에 기초를 두고 인가결정을 해야 하는 경우가 있다. 일곱 번째는 정보보안정책들은 인가결정을 하기 위해 주체, 자원, 행위의 속성들을 대상으로 연산을 할 때 사용가능한 연산자가 있다. 여덟 번째는 분산시스템을 위한 정책분산 지원이다. 개별적인 정책들은 다수의 정책 작성자에 의해 작성되며, 다수의 PEP에서 수행될 수 있다. 독립적인 정책들을 모으거나 조합하여 이용할 수 있을 뿐만 아니라, 필요에 따라 정책 갱신도 허용된다. XACML은 본 논문에서 구현하는 정책 평가 모듈의 구조에 적합하고 주체를 그 속성에 따라 식별 할 수 있다는 점에서 유연한 접근통제 정책 명세를 지원할 수 있는 장점이 있다.

III. 프라이버시 보호 정책 언어

프라이버시는 개인정보보호 보다는 보다 포괄적인 의미로 개인정보에 대한 소유자의 자기정보통제권이라는 측면이 강조된 것이다. 우리나라는 '정보통신망 이용촉진 및 정보보호 등에 관한 법률'에 정보 소유자의 자기 정보에 대한 통제를 명시하고 있다. 또 프라이버시 보호에 관해서 국외의 다른 기관들도 요구사항을 명시하고 있다. 대표적이 것은 OECD 프라이버시 보호 가이드라인, CSA 프라이버시 코드, EU 개인정보 보호지침, 아이덴티티 7법규 등을 예로 들 수 있다. 이와 마찬가지로 소셜 네트워크 서비스에서도 개인의 정보에 대한 권리를 주장 하는 움직임이 활발히 일어났으며 권리 장전을 발표하였다.

사용자 권리보호를 위해서 다음 세 가지 권리를 보

장해 주길 주장한다.

- 내 개인정보 등에 대한 "소유권(Ownership)"
- 내 개인정보 공유방법에 대한 "통제권 (control)"
- 내가 신뢰하는 외부 사이트가 내 개인정보에 접근할 수 있는 권한을 줄 수 있는 "자유권 (Freedom)"

그리고 다음 세 가지를 제공자의 의무로 제시한다.

- 사용자의 개인정보는 사용자가 원한다면 온라인에서 직접 출판할 수 있어야 하고 persistent URL 이나 API, open data format 등을 통해 서로 공유할 수 있도록 해야 한다.
- 타 사이트에서의 활동내역을 출판(syndicate)할 수 있도록 해야 한다.
- 사용자의 프로필 페이지에 외부 링크를 삽입할 수 있도록 해야 한다.
- 사용자들이 어느 사이트에서나 통용될 수 있는 식별자로 그들이 아는 사람을 찾을 수 있도록 해야 한다.

위의 권리 장전과 같은 사용자의 완전한 권리를 명시 할 수 있는 사용자 정책을 설정 가능하도록 해야 한다.

3.1 제안하는 정책언어

소셜 네트워크 서비스 사이트들의 접근통제 요구사항들을 만족 할 수 있는 접근통제 모델로 앞서 밝힌 것처럼 속성기반 접근통제 모델에 기반한 접근통제 정책언어를 제안한다. 이는 어떤 플랫폼에든 적용 가능할 수 있도록 XML 로 기술하며, 자원 요구자의 요청을 개인정보 소유자의 정책과 사용자의 개입 없이 비교하여 평가를 내리도록 설계한다. 또 속성기반 접근통제의 모델을 바탕으로 접근통제에 주체의 속성과 자원의 속성을 이용하여 보다 세밀한 접근통제를 달성한다. 그리고 OASIS의 접근통제 언어 표준인 XACML의 표준 데이터 타입과 다양한 함수를 받아들여 표현력을 증가시켰다. 또 일반 사용자가 정책을 명세할 때 정책의 충돌 회피까지 고려하기 힘듦을 반영하여 '허가' 사항만을 명세하도록 하여 정책의 충돌을 최소화 하고, 개인정보의 보호는 최대화 하고 있다.

접근통제를 위한 접근제어 모델은 일반적으로 '주체-액션-객체'의 구조를 갖는다. 미리 알려진 주체에 대한 정책을 설정하는 것으로 해당 주체가 객체에 어떤 연산을 가할 수 있는가를 기술하고 집행하는 것이다. 이와 같은 접근제어 정책의 구조를 바탕으로 사용의 목적과 의무사항 등을 규칙의 요소로 추가하여 프라이버시 보호 측면을 강화 할 수 있다.

(1)은 기본적인 접근제어 정책의 표현이며, (2)는 프라이버시를 보호하기 위해 변경 된 정책이다.

(1) (Subject, Permissions(Action, Resource))
(2) (Subject, Rule(Purpose, Permissions (Action, Resource), Obligation)

제안하는 소셜 네트워크 서비스의 사용자 정책의 경우 자원의 소유자가 자원에 대한 접근 통제 정책을 명세하는 것으로 접근권한을 다른 사용자나 접근을 원하는 에이전트에게 허가하거나 거부하는 방식이다. 따라서 사용자가 명세하는 접근통제 정책의 경우 주체는 사용자의 개인정보에 접근하고자하는 사람을 포함한 모든 것이 될 수 있다.

3.1.1 SNPL(Social Network Policy Language) 구성

SNPL의 구성요소는 [그림 1]과 같다.

E : 1
E? : 0 or 1
E+ : 1 or unbounded
E* : 0 or unbounded

[그림 1] 정책언어 표현 방법

엔티티명만 있는 경우는 해당 범위(scope)내에 한 번 기술되는 것이고, 엔티티명과 '?'가 함께 있는 경우 해당 범위 내에 기술되지 않거나 한 번만 기술 된다. '+' 기호와 함께 명시된 경우에는 한번 이상 범위 내에 기술 될 수 있으며 '*' 와 함께 명시 된 경우는 범위 내에서 전혀 사용되지 않거나 제한 없이 사용가능한 것이다.

위의 표현 방법을 사용하여 작성된 SNPL의 일반적인 구조의 예는 [그림 2]와 같다.

각 정책언어 요소를 살펴보면 다음과 같다. PolicySet은 여러 가지 Policy를 관리 할 수 있는 요소로써 PolicySetId를 통해 식별가능하며, 사용자별 PolicySet이 존재하기 때문에 이를 식별할 수 있도록 식별정보를 제공한다. PolicySet의 자식요소로는 Description, Policy, PolicyIdReference가 있다.

PolicySet(PolicySetId) = (Description*, Policy*, PolicyIdReference*)
Policy(PolicyId) = (Subject, PermissionSet)
Subject(SubjectId) = (SubjectMatch)
RuleSet(RuleSetId) = (Rule+, RuleIdReference*, RuleSetIdReference*)
Rule(RuleId) = (VariableDefinition*, Purpose, Permission, Condition?, ObligationSet?)
Permission(PermissionId) = (Resources, Action)

〔그림 2〕 SNPL 구성 요소

Policy는 사용자 정책을 직접 나타내는 요소로 PolicyId를 통해 식별가능하며, 사용자별 Policy를 식별할 수 있는 식별정보를 제공한다. Policy의 주요 자식요소는 Subject와 RuleSet이 있고 유연성을 부여하기 위해 SubjectIdReference, RuleSetIdReference가 존재한다.

Subject는 자원에 접근 가능한 엔티티에 대한 속성을 명시하는 곳이다. Subject의 자식 요소인 SubjectMatch를 통해 자원 요청자의 요청으로부터 요청자의 주체와 정책의 주체를 비교하여 정책의 주체의 속성과 요청자의 속성이 동일 한 가를 판단한다. SubjectMatch도 Id를 갖는데 이를 통해서 해당 속성을 비교할 수 있는 함수를 지시한다. 이 기능 함수는 OASIS의 XACML의 기능 함수 정의를 차용한다.

RuleSet은 주체에 배정되는 규칙들의 모음이다. RuleSet은 사용자가 명시한 사용자 규칙과 사이트의 관리자가 명시한 기본 규칙으로 나눌 수 있으며, RuleSetId를 통해 식별하고 자식 요소로는 Rule, RuleIdReference, RuleSetIdReference가 있다. RuleIdReference와 RuleSetIdReference를 통해서 다른 Rule과 RuleSet을 참조할 수 있어 유연함을 제공한다. 특히 관리자의 기본 규칙과 사용자의 규칙을 동시에 적용할 경우 유용하다.

Rule은 주체에 배정되는 규칙이다. 이는 자원을 이용하기 위해 적용되는 규칙이며 RuleId로 관리되고 식별 될 수 있다. Rule의 자식은 정책에서 유연성을 제공할 수 있는 VariableDefinition, 자원에 접근하는 목적을 명시하는 Purpose, 사용자에게 허가를 명시하는 Permission, 환경정보(context)와 사용자의 동의(consent)를 표현 가능한 Condition, 허가를 얻기 위해 강제하는 의무사항인 ObligationSet으로 구성된다.

Permission은 자원과 자원에 가할 수 있는 연산(자원에 가하는 행위)의 쌍으로 이를 허가 한다는 의미를 내포하고 있는 요소이다. PermissionId를 통해 관리될 수 있고 식별될 수 있다. Permission의 자식요소로는 Resources와 Action이 있으며 각각은 자원 요청자에게 접근을 허가하는 자원과 그 자원에 대한 연산을 의미한다. Resources는 Resource의 모음으로 구성되어지고, 정책에서 Action의 값으로는 read, write, delete, update 만 허용한다.

위의 요소들을 이용해서 사용자에 의해 발행되는 정책은 자원에 접근을 요청하는 주체의 속성이 정책에 명시한 주체 속성과의 매치를 통해 참이 되는지 여부를 판단한다. 그리고 규칙(Rule)의 각 사항들 역시 요청자가 제시하고 있는 목적, 자원, 연산이 합당한가를 매치를 통해 참과 거짓 여부를 판단한다. 이런 일연의 과정을 통해서 자원에 대한 접근과 자원 요청자의 행위를 통제할 수 있다. 또한 Qun Ni와 Bertino 등이 밝힌 프라이버시 강화요소인 목적(Purpose), 의무상항(Obligation), 컨디션을 이용한 사용자 동의(consent)를 정책의 요소로 넣음으로써 프라이버시 보호를 강화하고 있다[11].

또 소셜 네트워크 서비스의 핵심인 소셜 관계를 주체의 속성으로 사용하고 주체의 다양한 속성을 통해 누구에게 자신의 자원에 접근하는 것을 허가할 것인지에 대한 권한 부여의 판단이 가능하기 때문에 권한 수준의 유연성을 정책에 부여 가능하고 보다 세밀한 접근 통제가 가능하다.

만약 다른 소셜 네트워크 사이트에 접근했을 때 누구에게나 공유가 가능한 아이덴티티 자원이거나, 다른 소셜 네트워크 환경에 유지 되고 있는 자신의 개인정보 아이덴티티는 공유 운용하는데 현재 사이트들이 제공하는 접근 통제 방법으로 문제가 없을 것이다. 하지만 자신을 중심으로 접근이 가능한 친구나 그룹정보들을 다른 소셜 네트워크 서비스에 아이덴티티정보로 제공하는 것에 관한 문제는 어려운 문제다. 이를 해결할 수 있는 방법은 개인정보 공유의 각 상황을 모두 고려하고 개인정보 접근 및 이관에 대한 정책파일을 생성하고 이 정책에 기반을 두어 인가를 수행해야 한다. 또 하나이상의 소셜 네트워크 사이트에 가입한 사용자라면 소셜 사이트들을 연계하여 새로운 커뮤니티를 형성함으로써 사용자들 간에 공유운용 할 개인정보를 새롭게 설정함으로써 소셜 사이트를 통합한 새로운 가상 소셜 네트워크 서비스 사이트의 구축도 생각해 볼 수 있다. 단, 이 경우에는 연관된 사용자들의 개인정보에 대한 공유를 동의할 수 있는 메커니즘과 제반 정책이

요구된다. ABAC을 적용한 SNPL을 이용하여 접근 정책을 기술하고 이를 시스템에 적용하면 이 역시 해결 가능하다.

3.1.2 SNPL 예

사용자는 자신의 프라이버시 정책을 가질 수 있으며 이를 SNPL로 표현하는 사례는 다음과 같다.

철수라는 사용자가 친구 영희에게 자신의 Activity를 공유하고자 한다.

철수의 프라이버시보호를 위해 "email이 abcd@example.com을 사용하는 친구인 영희에게 회의 목적으로 일정 데이터를 공유한다." 라는 정책을 세웠다. 이 때 Activity를 공유하기 위해 세울 정책은 주체의 속성으로 abcd@example.com, Friend, 영희 등을 갖게 되고, 정책의 판단의 결과는 'Permit' 일 경우에만 해당 데이터를 열람할 수 있다. 또한 데이터에 가능한 연산은 'read'가 되고 열람 할 자원은 schedules라는 속성을 갖는다. 그리고 열람의 목적은 meeting이 된다. 이를 표현한 SNPL 예제는 부록1과 같다.

SNPL은 정책을 표현하는 방법뿐만 아니라 요청을 정형화 시키는 방법도 함께 제안한다. 부록1 정책에 의해 데이터 연산에 대한 허가를 받을 수 있는 주체, 액션, 자원, 목적을 포함한 요청의 예는 부록2와 같다.

IV. 프라이버시 보호 모듈 설계 및 구현

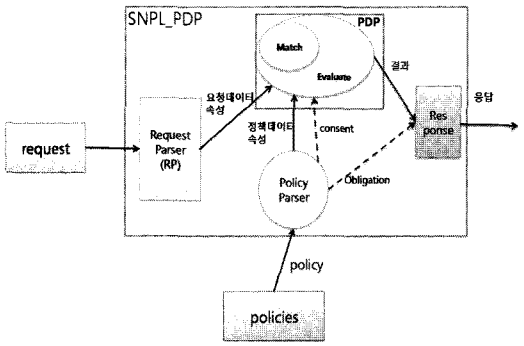
4.1 정책 결정점 설계 및 구현

지금까지 소셜 네트워크 서비스들 간 정보의 공유에 있어서 필요기술과 프라이버시보호 및 개인정보보호에 대한 요구사항들에 대해 알아보았다. 그리고 각 항목에서의 요구 사항들을 해결 하기위한 방법으로 프라이버시 보호가 강화된 접근통제 정책언어인 SNPL을 제안하였다.

제안한 SNPL로 작성한 정책을 시스템에 적용하기 위해 언어파서와 요청에 대한 정책 평가 결과를 내는 정책 결정점(Policy Decision Point, PDP)이 필요하며 그 구성요소와 동작 방식은 [그림 3]과 같다.

각 구성 요소와 역할을 살펴보면 다음과 같다.

SNPL_PDP: 모든 구성 요소들을 포함하고 있는 요소로 동일 이름의 클래스로 후에 시스템에 존재한다.



[그림 3] SNPL 정책 결정점(PDP) 구성요소

policies: 정책들을 모아놓은 것으로 파일 시스템을 그대로 이용할 수도 있고 관리의 효율을 높이기 위해 DBMS를 사용할 수도 있다. 구현하는 시스템에서는 파일 시스템에서 policy 디렉토리 아래의 모든 '.xml'파일들을 말한다.

PolicyParser: policy 파일을 위한 파서이다. 구현하는 시스템에서는 jnu.ssrc.snpl 패키지의 PolicyFileStore 클래스로 모든 정책파일들이 파싱시키는 역할을 한다.

request: 사용자의 요청은 컨텍스트 스키마에 유효한 형태의 XML로 시스템 내에서 만들어지는데 이것이 request이다. 현재는 도메인에 한정된 입력 값들을 XML문서로 표현한 request폴더 아래에 request.xml 파일로 놓인다.

RequestParser(RP): XML 형식인 request를 파싱하는 역할을 한다. 현재는 SNPL_PDP에 입력으로 requset.xml 파일이 들어오면 jnu.ssrc.snpl.ctx 패키지의 RequestCtx 클래스에 의해서 request라는 변수에 모든 요소가 파싱되어 PDP가 정책과 비교할 수 있는 형태가 된다.

PDP: 실제로 정책과 요청간의 평가(Permit, Deny)를 내리는 요소이다. 동일 이름의 클래스로 요청에 대해 evaluate 메소드를 사용하여 사용자의 요청이 정책에 위배되는 사항이 없는지 평가를 내린다.

Match: 요청에 대한 정책과의 비교를 위해서 사용하는 boolean 타입의 메소드이다. PDP의 evaluate에서 가장 먼저 불리어지며, 실제로 매치의 과정은 jnu.ssrc.snpl.parsing.type 패키지 내의 RoleMatch, PurposeMatch, PersonalDataMatch, ActionMatch라는 클래스를 통해 이루어진다. 클래스 모두 Match 메소드를 사용하고, 정책에 쓰여 진 순서에 따라 각 요청 요소들을 정책과 비교한

다. 모든 과정이 True이면 해당 요청을 허가할 수 있다. Match 과정 중에 한 요소라도 반환 값이 False이면 진행 중인 평가과정을 중지하고 다음정책을 평가하도록 구현되어 있다.

Result: Match에 의한 비교과정이 끝난 결과를 가지고 허가나 불가의 결정을 내려주는 역할을 한다. 이것은 result 객체의 값이 true인 경우 결과가 'Permit'이고 false이면 'Deny'이다. 이때 해당 정책에 Condition이 있다면 확인하여 결과에 영향을 주도록 한다.

response작성기: Result의 결과를 이용하여 소셜 네트워크 서비스에게 xml 형태의 응답을 제공하는 역할을 한다. 이때 의무사항(Obligation)이 존재하는지를 확인 한 후 존재하면 이를 포함시켜서 xml형태로 내보낸다.

4.2 소셜 네트워크 서비스 환경 구축

소셜 네트워크 환경을 구축하기 위해 공개 소프트웨어에 기반을 둔 Shindig라는 오픈소스를 사용하였다[12, 13]. 이를 다운로드 받은 후, 커스터마이징을 통해 개발한 프라이버시 모호 모듈이 서비스 될 소셜 네트워크 서비스를 구축하였다.

구축된 shindig와 partuza서비스에 사용자의 정책을 반영할 수 있도록 정책을 작성하는 기능을 [그림 4]와 같이 makeProfile 탭에 구현하였다.

사용자는 서비스 사이트의 프로파일 작성 화면의 Make Policy 탭을 클릭하고 구성요소의 항목에 값을 넣고 화면 하단의 save버튼을 클릭하면 정책 발행 화면에 사용자가 입력한 값들은 서버의 process.php에 전달되고 "policy_[random수].xml" 형태로 만들어진다. [그림 5]는 이 새로 사용자 정책이 생성된 것을 보여 준다.

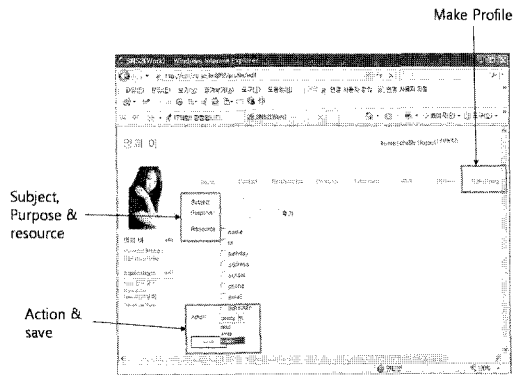
이렇게 구축된 구성 요소들을 바탕으로 실행되는 서비스의 절차는 다음과 같다.

① 소셜 네트워크 서비스의 소셜 가젯을 통해서 개인정보 요청자는 서비스 제공자의 입장이 되는 소셜 네트워크 서비스에 데이터 요청을 전송한다.

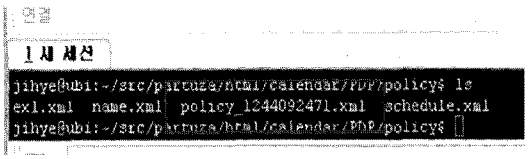
② 요청을 받은 서비스는 요청자의 인증 정보를 요구한다.

③ 요청자는 증명정보와 요구하는 자원에 대한 정보를 전송한다.

④ 자원을 제공할 서비스는 이를 바탕으로 요청 컨텍스트를 작성하고 이 요청문을 SNPL 모듈에 전송한다.



[그림 4] 사용자 정책 발행



[그림 5] 생성된 사용자 정책

⑤ SNPL 모듈은 요청을 정책과 평가한 후 인가 결정을 내보낸다.

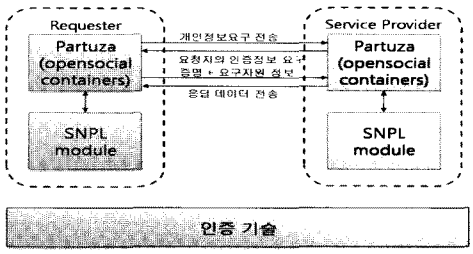
⑥ 인가 경정에 따라 데이터를 요청 받은 사이트는 응답 데이터를 전송한다.

인증 부분은 대부분의 사이트들이 자신들의 방법으로 서비스를 하고 있으며 현재 서비스들 사이에 데이터를 공유하기 위해서 가장 널리 쓰이는 방법은 OpenId이다. OpenId는 특정 서비스 제공 사이트에 종속되지 않고 인증서비스를 제공할 수 있는 기능을 제공하기 때문에 서로 다른 사이트들끼리 충분히 서로의 인증을 할 수 있다.

4.3 비교분석

본 논문에서 제안한 SNPL은 데이터에 접근하는 주체의 속성에 기반 해 소셜 네트워크 서비스 사이에 데이터를 공유 할 때 데이터에 대한 접근통제 정책을 기술하는 방법을 제공한다. 프라이버시 보호 요소인 목적, 조건, 의무사항 등의 항목을 표현 가능하도록 함으로써 프라이버시가 강화된 데이터 공유를 가능하게 한다.

[표 1]은 다른 접근 통제 정책 언어들과 제안하는 모델에 사용하는 접근 통제 언어인 SNPL을 비교한 것이다.



〔그림 6〕 자동화된 정책 기반 인가 구조

〔표 1〕 다른 정책 언어와 SNPL 비교

비교대상	주체 식별방법	프라이버시 고려	정책설정 주체	소셜 관계 표현
SNPL	속성	O	자원소유자	O
EPAL	user-category	O	보안관리자	X
XACML	속성	O	보안관리자	O
P3P	법적 정책 고지자	O	보안관리자	X

소셜 네트워크 서비스의 특징은 개인화에 있다. 따라서 정책도 소유자 개인이 자신의 의도대로 설정할 수 있어야 한다. 하지만 다른 정책 언어들은 대부분이 기업 환경에서 개인정보를 수집하고 이를 이용하는데 필요한 정책을 명세하기 위해 만들어진 것으로 이러한 특징을 잘 반영하지 못하고, 이를 사용하는 시스템들은 기업 환경에서 보안관리자에 의해 정책이 명세 된다는 것을 알 수 있다.

소셜 네트워크 서비스에서 자원에 접근하고자 하는 주체는 개인이 되어야한다. 어떤 기업의 역할에 따라 구분된 주체는 아니다. 따라서 주체의 식별 방법에 유연성이 필요하며, 이를 위해 SNPL은 ABAC이라는 속성기반 접근 통제 모델을 사용하여 주체를 주체의 속성에 따라 식별하고 있다. 하지만 EPAL은 기업의 역할에 맞는 주체의 신분을 미리 정해진 단어집에 의해 식별하고, P3P는 정책을 고지하는 법적인 주체를 명시하여 소셜 네트워크 서비스와는 전혀 어울리지 않음을 알 수 있다. 당연한 이야기지만 소셜 네트워크 서비스에서 사용자의 소셜 관계 표현과 관리는 중요한 부분이다. 따라서 정책의 유연성이나 접근통제의 유연성을 위해 소셜 관계를 사용하는 것은 당연하다고 할 수 있다. 그런데 다른 정책 언어들은 이를 지원할 수 있는 방법을 제공하지 않는다.

XACML의 경우 주체, 자원 등의 속성을 표현 가능하여 이를 표현하는데 사용가능하고 범용적으로 만들어진 정책 언어라 대부분의 접근통제 모델을 표현 가능하다. 하지만 이 범용성은 정책을 명세하고 관리하는데 어려움을 주며, 사용자가 스스로 자신의 정책을 명세하기에는 아직 잘 만들어진 정책 작성기가 없어 매우 어렵다. 또한 XACML은 본 논문에서 제시하는 정책 언어보다 무겁고 일반 사용자가 정책을 명세하는데 정책 충돌, rule의 규칙 등 고려 사항이 많은 단점이 있다.

V. 결 론

소셜 네트워크 서비스는 사용자의 참여, 공유, 개방 등 웹 2.0의 핵심 가치가 가장 잘 들어나는 서비스이며, 이를 통해 다양한 콘텐츠를 확보하여 새로운 형태의 웹 서비스의 모양을 보여 주었다. 하지만 서비스 사이트들이 사용자의 개인정보를 거대한 silo 형태로 유지하고 사용자의 자기 정보에 대한 권리를 보장해 주지 않고 있다. 또 사용자는 닫힌 형태의 서비스 모델들 때문에 실세계의 소셜 관계를 온라인에 완벽하게 구축할 수 없고 다른 서비스 사이트 사용자와 정보를 공유하기 위해서 각 사이트마다 가입을 하고 새롭게 소셜 관계를 구축해야 한다. 이는 개인정보의 산재와 더불어 정보의 불일치, 또 개인정보 소유자의 권리에 대한 침해 등 잠재적 프라이버시 문제에 대한 제기를 받고 있다. 이에 따라 몇몇 소셜 네트워크 서비스 사이트들은 데이터를 공유할 수 있는 APIs을 공개함을 통해 부분적으로 데이터 공유를 지원하고 있다. 본 논문에서는 서비스 간 데이터 공유가 가능한 SNS 모델을 보여주고, 개인 정보를 재사용 가능하게 했다. 이 과정에서 소유자에게 자기 정보에 대한 통제권과 소유권을 보장함으로써 개인 정보 소유자의 권리를 보장하는 방법으로 정책에 기반한 프라이버시가 강화된 접근 통제를 가능하게 하는 프라이버시 보호 접근통제 모듈을 제안했으며, 시스템에 반영할 수 있는 정책을 명세하는 SNPL이라는 정책언어를 제안했다.

제안한 접근통제 모듈과 SNPL은 어떤 플랫폼이든 상관없이 채용할 수 있도록 각각 자바로 구현하고 XML로 기술했으며, 어떤 소셜 네트워크 서비스 사이트라도 채용하여 사용 가능하다. 그리고 SNPL은 속성기반 접근통제 모델을 기반으로 하여 데이터에 접근해서 오퍼레이션을 하기 원하는 주체에 대해 주체의 속성을 통해서 접근 권한을 인가하는 방법을 제안한다. 주체의 속성은 주체를 식별하거나 주체의 특성을

표현 할 수 있는 모든 것을 포함하며, 소셜 네트워크 서비스의 특징인 소셜 관계를 속성에 포함하여 그 범위를 넓혔다. 또 이를 통해 접근 통제의 수준을 결정할 수 있어 보다 세밀한 접근 통제가 가능하고 소유자가 직접 공유 정책을 작성하고 소유자가 원하는 다른 사이트나 사용자와 데이터를 공유 가능하게 하여 소유자의 의도를 반영한 접근 통제 방법을 제공함으로써 소유자의 자기 정보에 대한 권리를 높였다. 정책에 프라이버시 강화 요소를 반영해서 소유자의 권리 보장과 프라이버시 강화 요소들에 의해 프라이버시 보호 측면도 강화했다.

웹은 계속해서 진화하고 있다. 아직 산업화 단계는 아니지만, 의미기반의 웹(데이터 웹)에 대한 연구들이 이미 어느정도 진행되어 요소기술들이 개발되고 있다. 이런 발전 방향에 맞춰 소셜 네트워크 서비스 역시 의미기반으로 발전할 것이며, 이와 같은 환경에서 데이터의 공유는 당연시 될 것이다. 이런 완전히 개방된 환경에서 개인정보 소유자의 프라이버시 문제는 더욱 대두 될 것이며 본 논문의 제안 요소들을 이 분야에도 적용할 수 있으리라 예상된다. 또한 속성기반 접근통제 모델을 반영하는데 있어 의미를 기반으로 하는 추론 과정이 없을 경우 요청자의 요구 명세서와 정책 명세서 사이 주체의 속성 매치에는 한계가 있다. 따라서 각 요소를 온톨로지 기술언어를 사용해 재정의 하고 의미기반의 웹의 서비스에서도 프라이버시를 강화한 접근통제를 위한 정책 언어로 사용 가능하도록 계속 발전시키는 연구가 필요하다.

참고문헌

[1] Ching-man Au Yeung, "Decentralization: The Future of Online Social Networking," W3C Workshop on the Future of Social Networking, Position Papers, Jan. 2009.

[2] Maria Aspan, "How Sticky Is Membership on Facebook? Just Try Breaking Free," The New York Times, Feb. 2008.

[3] Kevin Poulsen, "Pillaged MySpace Photos Show Up in Massive BitTorrent Download," Wired, Jan. 2008.

[4] Jeremiah Owyang, "The Many Challenges of Social Network Sites," Web Strategy blog, Feb. 2008.

[5] W3C, "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification," http://www.w3.org/TR/P3P/, April 2002.

[6] C. Shankar and R. Campbell, "A policy-based management framework for pervasive systems using axiomatized rule-actions," Proceedings of the Fourth IEEE International Symposium on Network Computing and Applications, pp. 255~258, 2005.

[7] IBM, EPAL v1.2, http://www.zurich.ibm.com/security/enterprise-privacy/epal/Specification/index.html

[8] OASIS, "eXtensible Access Control Markup Language(XACML) V2.0," Committee draft 04, Dec. 2004.

[9] E. Yuan and J. Tong, "Attributed based access control (ABAC) for Web services", ICWS 2005 Proceedings, 2005 IEEE International Conference.

[10] T. Priebe, E.B. Fernandez, J.I. Mehlau, and G. Pernul, "A Pattern System for Access Control," Research Directions In Data And Applications Security XVIII: IFIP TC 11/WG 11.3 Eighteenth Annual Conference On Data And Applications Security, Jul. 2004.

[11] Qun Ni, "Privacy-aware Role Based Access Control," SACMAT'07, Jun. 2007.

[12] Shindig, http://incubator.apache.org/shindig/

[13] Shindig, http://www.truveo.com/Google-IO-2008-Apache-Shindig/id/3310752665

```
<?xml version="1.0" encoding="UTF-8"?>
<PolicySet PolicySetId="younghee:policySet:1" xmlns="urn:jnu:ssrc:sns:1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:jnu:ssrc:sns:1.0 Policy.xsd">
 <Policy PolicyId="urn:snsl:policy:1" issuer="cheolsoo">
   <Subject>
      <SubjectMatch  MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
          abcd@example.com
        </AttributeValue>
        <SubjectAttributeDesignator AttributeId="urn:snsl:subject:subject-id:1"
        DataType="http://www.w3.org/2001/XMLSchema#string"/>
      </SubjectMatch>
      <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
          Friend
        </AttributeValue>
        <SubjectAttributeDesignator AttributeId="urn:snsl:subject:subject-id:2"
        DataType="http://www.w3.org/2001/XMLSchema#string"/>
      </SubjectMatch>
      <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
          Younghee
        </AttributeValue>
        <SubjectAttributeDesignator AttributeId="urn:snsl:subject:subject-id:3"
        DataType="http://www.w3.org/2001/XMLSchema#string"/>
      </SubjectMatch>
    </Subject>
    <RuleSet RuleSetId="urn:snsl:ruleset:id:1" issuer="cheolsoo">
     <Rule RuleId="urn:snsl:rule:id:1" issuer="cheolsoo">
      <Purpose>
          <PurposeMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string"> meeting</AttributeValue>
          <PurposeAttributeDesignator DataType="http://www.w3.org/2001/XMLSchema#string"
         AttributeId="urn:snsl:purpose:purpose-id"/>
       </PurposeMatch>
      </Purpose>
     <Permission PermissionId="urn:snsl:permission:younghee:id:1">
      <Resources>
       <Resource>
        <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
         <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
          scheduledata
         </AttributeValue>
         <ResourceAttributeDesignator Issuer="abcd@abcd.com" DataType="http://www.w3.org/2001/XMLSchema#string"
                 AttributeId="urn:snsl:resource:resource-id"/>
        </ResourceMatch>
       </Resource>
      </Resources>
      <Action>
       <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
         read
        </AttributeValue>
        <ActionAttributeDesignator DataType="http://www.w3.org/2001/XMLSchema#string"
                 AttributeId="urn:snsl:action:action-id"/>
        </ActionMatch>
       </Action>
      </Permission>
     </Rule>
    </RuleSet>
  </Policy>
</PolicySet>
```

〔부록1〕. 정책의 SNPL 형식

```
<?xml version="1.0" encoding="UTF-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:jnu:ssrc:snplctx:1.0 PolicyCtx.xsd" xmlns="urn:jnu:ssrc:snplctx:1.0">
  <Subject>
    <Attribute AttributeId="urn:snsl:subject:subject-id:1" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>abcd@example.com</AttributeValue>
    </Attribute>
     <Attribute AttributeId="urn:snsl:subject:subject-id:2" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>Friend</AttributeValue>
    </Attribute>
    <Attribute AttributeId="urn:snsl:subject:subject-id:3" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>Younghee</AttributeValue>
    </Attribute>
  </Subject>
  <Resource>
    <Attribute AttributeId="urn:snsl:resource:resource-id" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>
          scheduledata
        </AttributeValue>
    </Attribute>
  </Resource>
  <Action>
    <Attribute AttributeId="urn:snsl:action:action-id" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>read</AttributeValue>
    </Attribute>
  </Action>
  <Purpose>
    <Attribute AttributeId="urn:snsl:purpose:purpose-id" DataType="http://www.w3.org/2001/XMLSchema#string">
      <AttributeValue>scheduling</AttributeValue>
    </Attribute>
  </Purpose>
</Request>
```

〔부록2〕. 요청의 SNPL 형식

〈著者紹介〉

김 지 혜 (Ji-hye Kim) 정회원
2006년 2월: 조선대학교 컴퓨터공학과 학사
2009년 8월: 전남대학교 정보보호학과 석사
2010년 3월~현재: 전남대학교 정보보호학과 박사과정
2010년 7월~현재: 한국인터넷진흥원
〈관심분야〉 프라이버시보호, Identity 관리시스템, SNS 개인정보보호

이 형 효 (HyungHyo Lee) 종신회원
1987년 2월: 전남대학교 계산통계학과 학사
1989년 2월: KAIST 전산학과(석사)
1990년 2월: 전남대학교 대학원 전산학과(박사)
1990년~1992년: 삼보컴퓨터 기술연구소
1993년~1997년: 한국통신 연구개발원
2001년 3월~현재: 원광대학교 정보·전자상거래학부 부교수
〈관심분야〉 프라이버시보호, Identity 관리시스템, 보안 온톨로지, 응용보안