# 지능적인 Identity 발견 서비스 설계 및 구현*

전은국*○, 박희만*, 이영록*, 이형효**, 노봉남*
* 전남대학교, ** 원광대학교
*{livingstoneO, hareup, dogu, bongnam}@lsrc.jnu.ac.kr, **hlee@wonkwang.ac.kr

# Design and implementation of Smart Identity Discovery Service

EunGook Chun*○, HeeMan Park*, YoungLok Lee*, HyungHyo Lee**, BongNam Noh*
* Chonnam University, **WonKwang University

**요 약**

최근 인터넷 상의 개인정보의 유출로 인한 피해가 급증하면서, 인터넷 상에서 사용하는 사용자 정보 관리 문제가 크게 대두되고 있다. 이들을 해결하기 위하여 여러 Identity 관리 시스템이 등장했다. 이들 중 OpenID는 사용자 중심의 Identity 관리 기술이며, url을 기반으로 하는 사용자 정보 제공 시스템이다. 기존의 웹 서비스는 입력된 사용자의 정보를 각각 저장하여 관리하는 반면, OpenID는 OpenID를 지원하는 특정 사이트를 지정함으로써 그곳에서 사용자의 정보를 제공하는 것이 특징이다. 하지만 이런 분산화된 사용자의 정보는 웹서비스에서 필요하는 사용자에 정보에 따라 각각 다른 url를 입력해야 한다. 이와 같은 방법은 기존의 각 사이트마다의 아이디와 패스워드를 알아야 하는 것과 같은 현상을 초래할 수 있다.

본 논문을 통해 구현한 서비스는 웹 서비스에서 요구하는 사용자의 정보에 따라 동적으로 사용자 정보 제공자를 선택하거나 필요한 정보만을 요청함으로써 하나의 url만으로 원하는 모든 웹 서비스를 이용할 수 있도록 한다.

## 1. 서 론

인터넷의 확산과 웹 2.0 환경의 도래에 따라, 사용자가 관리해야 하는 디지털 형태의 Identity 정보가 기하급수적으로 증가하고 있다[1]. 즉, 현재 제공되고 있는 국내외 대부분의 웹 사이트에서는 사용자의 개인 정보를 요구하고 있다. 이러한 것은 사용자의 정보들이 산재됨을 초래하며, 심지어 사용자는 자신이 어느 곳에 가입을 했는지도 잘 기억하고 있지 않는다.

인터넷 사용의 확대와 편리성으로 사용자가 자신의 정보를 제공한 것에 대해 민감해 하지 않는 점을 틈타, 현재 많은 곳에서 이를 악용하는 사례가 많이 발생하고 있다. 실제 입력된 사용자 정보에 대한 사용상의 규약인 약관을 자세히 살펴보는 사용자는 극히 드물 것이다.

또한 사용자가 등록한 개인정보는 서비스 제공자의 관리 부실 또는 해킹등에 의해 유출이 되는 경우도 있으며, 제공되는 웹 서비스의 잘못된 로봇배제(Robots Exclusion) 표준 준수로 원하지 않은 사용자들의 정보 노출을 초래 할 수 있다.[2].

따라서 사용자가 인터넷을 사용함에 있어 보다 편하고 안전한 환경을 제공하며, 사용자의 Identity 정보를 관리해 주는 Identity 관리 시스템이 등장하게 되었다. 본 논문에서는 이런 Identity 관리 시스템의 발전 동향을 소개하고, 현재 사용되는 사용자 중심의 Identity 관리 기술인 OpenID를 응용하여 보다 편리하고 안전한 사용자 중심의 Identity 관리 방법을 제안한다.

본 논문의 2장에서는 Identity의 대한 정의를 살펴보고, Identity 관리 시스템과 관련된 표준 및 기술을 소개하고 제안하는 서비스의 기반이 되는 OpenID에 대해 살펴본다. 3장에서는 제안하고자 하는 지능적인 Identity 발견 서비스를 설계 및 구현하며, 4장에서는 분석 및 향후 연구계획을 제시한다.

* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음
(IITA-2006-C1090-0603-0027)

## 2. 관련연구

### 2.1 Identity

인터넷 상의 여러 웹 사이트들은 사용자에게 수많은 서비스를 제공하고 있다. 사용자들은 서비스를 제공받기 위하여 각 사이트에 개인 정보를 등록하고 로그인 과정을 수행한다. 웹 사이트는 아이디와 패스워드 그리고 개인의 주소, 전화번호, 직업, 심지어 주민등록번호까지 많은 정보를 보관한다. 이러한 사용자 개인을 표현하는 정보들의 집합을 Identity(또는 Digital Identity)라 하고, 각각의 정보를 속성(Attribute)이라 한다[3].

기존의 웹 사이트들은 각자의 Identity 정보들을 저장하고 있어 동일 Identity들이 불필요하게 산재되어 있다. 그만큼 Identity 정보들은 공격자들에게 많은 부분을 노출하고 있는 것이며, 사용자의 인지없이 웹 서비스의 내부적인 연계로 Identity 정보들을 노출시키는 경우도 빈번하다.

이러한 보안상의 취약점을 방지하고자 사용자는

Identity 정보들이 저장된 웹 사이트 마다 각각의 보안 정책에 신경을 써야하는 입장이다.

### 2.2 Identity 관리 시스템의 발전

개인의 Identity 정보를 불법적인 접근으로부터 안전하게 보호하며, 보다 편리하게 웹 서비스를 제공받고자 Identity 관리 시스템이 등장하였다. Identity 관리 시스템은 다음과 같은 형태로 발전하였다.

- Silo

현재 사용하는 대부분의 웹 사이트 형태로서, 사용자의 Identity 정보들이 각 사이트에 저장되어 있는 형태이다. 이러한 환경에서 사용자의 Identity 정보는 각 웹 사이트별로 Identity가 중복적으로 존재하여 유출 및 도난의 위험이 있고, 서비스를 받기 위해 매번 사이트 방문시 인증해야 하는 번거러움이 있다[4]. 또한 Identity들이 고립되어 공유나 교환이 되지 않는다.

- Centralized Identity 관리

Silo 형태에서 시스템이 가지고 있는 Identity 정보를 서로 공유하는 것이 필요해짐에 따라 분산된 Identity 정보를 중앙 집중적으로 관리하는 모델이 고안되었다. 이 모델을 시작으로 SSO(Single Sign On)이 가능하게 되었다. 가능 대표적으로 Microsoft의 .Net Passport[5] 서비스이다. 하지만 이러한 모델은 Identity 정보의 집중으로 인한 공격자의 대상이 되기 쉽고, 기술의 독점화로 사실상 제공하는 웹사이트의 협력 업체 사이에서만의 정보공유가 이뤄져 활성화 되지 못했다.

- Federated Identity 관리

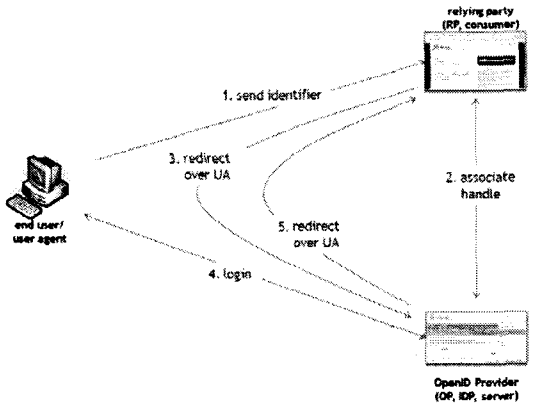
Centralized 모델의 단점을 보완하여 각 사이트의 정보 독립성을 존중하면서 사이트들 간에 정보를 공유할 수 있는 연결고리를 만들어 사용자에 대한 인증 정보 또는 Identity 정보를 공유한다. Liberty Alliance[4] 표준에 기반한 Identity 관리 시스템이 대표적이며, 2007년 현재 Liberty Alliance 표준에 기반한 Identity와 장치들이 정보, 교육 의료 등의 다양한 분야에 10억 개가 넘게 적용되고 있다.

- User-Centric Identity 관리

사용자의 정보 유출에 민감해지면서 최근 급부상한 사용자 중심 Identity 관리 모델은 사용자가 직접 자기가 사용할 인증방법과 인증서버를 선택하고, 실제 Identity 공유시 정보의 흐름이 사용자를 거쳐서 흘러가게하여 사용자의 Identity 정보에 대한 프라이버시 보호를 목적으로 하고 있다. 대표적인 사용자 중심 Identity 관리 시스템으로는 OpenID[5], LID[6], YADIS[7]를 들 수 있다.

### 2.3 OpenID

MS사의 .NET Passport시스템이 중앙 집중화된 구조인데 반해, OpenID 시스템은 사용자가 자신이 이용하려는 각 사이트마다 회원가입을 하여 계정을 갖지 않더라도 자신이 원하는 OpenID 제공자를 지정하여 필요한 속성만을 제공하여 서비스를 이용할 수 있는 URL기반의 Identity 관리 시스템이다.




(그림 1) OpenID 동작 흐름도

사용자 중심의 Identity 관리 모델의 선두주자격인 OpenID는 표준 스펙 자체가 공개되어 있기 때문에 어떤 서비스에서도 쉽게 OpenID 로그인을 지원할 수 있으며, 많은 OpenID 관련 라이브러리가 공개되어 있기도 하다. "OpenID Authentication 2.0"을 주축으로 규정되고 있는 OpenID 2.0 스펙은 인증 및 ID 관리 전반을 다루고 있으며, Identity Attribute에 대한 정의와 구성, 속성 교환(Attribute exchange)등을 위한 메시지(Message)들을 정의하고 있다.

(그림 1)는 OpenID 1.0에서 정의된 인증과정이다.

먼저 사용자가 원하는 서비스를 받기위해 선택한 RP(Relying party, consumer)에 서비스를 요청하고, 만약 RP에 사용자가 인증되지 않은 상태이면, RP는 사용자 식별자를 요청한다. 사용자가 자신의 사용자 식별자를 RP에게 전달하면, RP는 사용자식별자를 통해 OP(OpenID Provider, IdP or server)를 확인하고 OP와 associate과정을 거쳐 자신과 OP 간의 세션, 암호 키 등과 같은 공유 암호를 설정한다. 이 과정이 종료되면, RP는 사용자 브라우저를 경유하여 OP에게 인증을 요청한다. OP는 사용자가 인증되지 않았을 경우 사용자를 인증하고, 사용자의 인증 사실을 사용자 브라우저를 경유하여 RP에게 전달한다. RP는 사용자 인증정보를 확인하고 사용자에게 서비스 제공 유무를 결정한다.

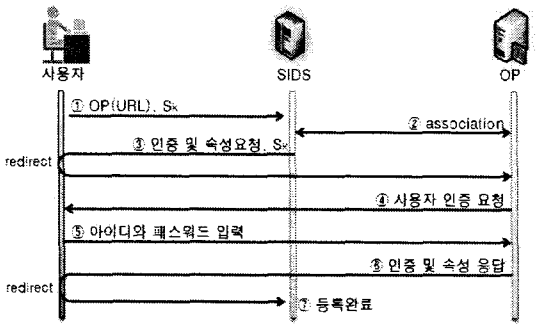
### 2.4 YADIS(Yet Another Decentralized Identity Service)

OpenID 1.0에서 정의한 단순 Html을 이용한 방법에서 발전한 YADIS는 Identity 제공자를 찾기 위한 Discovery 기능을 정의하고 있다[7]. YADIS 문서의 스키마는 XRDS와 XRD 스키마를 포함하고 있으며, 주요 기능으로는 Identity 제공자들의 리스트를 알려주는 것은 물론, 여

러 제공자에 대한 제공 우선순위를 부여할 수 있다.

## 3. SIDIS(Smart Identity Discovery Service)

구현하는 SIDIS에서는 다음과 같은 기능들을 통하여 보다 편리하고, 향상된 사용자 중심의 Identity 관리를 가능하게 한다.




(그림 2) OP 등록

### 3.1 OpenID 제공자 등록

사용자는 자신의 속성정보가 저장되어 있는 OP들을 등록한다. (그림 2)는 OP 등록 과정을 보여준다. 사용자는 OP를 가리키는 URL과 함께 OP와의 인증을 위한 키를 입력하면, SIDIS는 정상적으로 OP와 association과정을 거친 후 <표 2>와 같은 파라메터를 설정하여 사용자 속성을 요청한다.

<표 3> 사용자 속성 리스트 요청 포멧

| 파라메터 | 값 |
|---|---|
| openid.sidis.mode | request_attribute_list |
| openid.sidis.sid | 사용자가 입력한 아이디 값 |

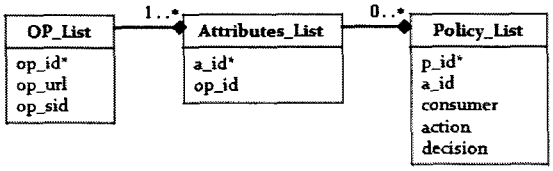
인증과 사용자의 속성 리스트를 요청받은 OP는 사용자로부터 아이디와 패스워드로 사용자임을 확인하고, 다음과 같은 포맷으로 요청에 대한 응답을 수행한다. OP는 사용자에 대한 속성들을 ','로 구분하여 "openid.sidis.attributes"에 설정된다.

<표 4> 사용자 속성 리스트 응답 포멧

| 파라메터 | 값 |
|---|---|
| openid.sidis.mode | response_attribute_list |
| openid.sidis.attributes | 입력된 사용자의 속성 리스트 |

### 3.1 속성에 대한 정책 설정

(그림 3)은 SIDIS에 구축된 데이터베이스들의 관계로, OP에 대한 정보와 각 OP가 가지고 있는 속성들, 그리고 그 속성들마다의 정책(policy)들의 관계를 나타내고 있다.

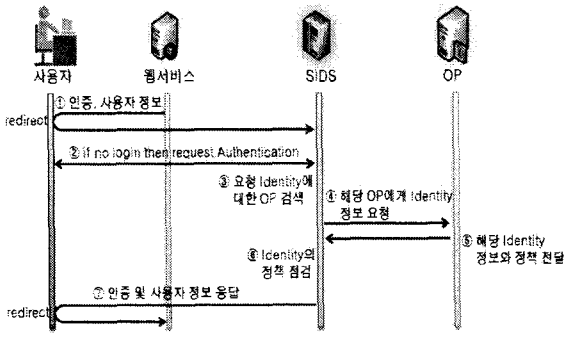



(그림 3) SIDIS 데이터베이스 구조 및 관계

정책에 대한 설정은 사용자가 SIDIS에 직접 접속하여 이루어지며, 각 속성(a_id)에 대해 사용되는 웹서비스(consumer)와 요청되는 특정한 동작(action; 생성, 읽기, 수정, 삭제)에 따른 결정(decision;허락,거절)을 지정한다.

### 3.2 사용자 정보 제공

구현된 서비스에서 사용자의 대한 정보 요청은 OpenID 2.0 스팩을 준수한다. (그림 4)은 사용자에 대한 속성을 제공해주는 절차를 보여준다.




( 그림 4 ) 사용자 속성 제공

사용자가 서비스를 받기위해 웹에 접근하면, 웹 서비스는 사용자에 대한 인증정보와 몇가지 사용자에 대한 속성을 요구한다. SIDIS는 요청한 사용자의 속성 정보에 따라 요구할 OP를 검색하고, 각 OP에 해당하는 속성정보들을 ','로 구분하여 "openid.sidis.requied" 파라메터를 설정한 후 OP에 전달한다.

<표 5> 사용자 속성 값 요청 포멧

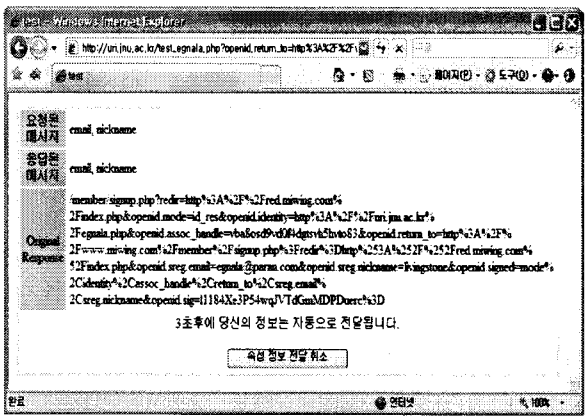
| 파라메터 | 값 |
|---|---|
| openid.sidis.mode | request_attribute_value |
| openid.sidis.sid | OP의 인증을 위한 값 |
| openid.sidis.requied | 요청된 사용자 속성들 |
| openid.sidis.request_count | 요청된 사용자 속성 갯수 |

사용자의 속성 값을 요청받은 OP는 sid값으로 사용자를 구분하여 해당 요청된 속성값을 (표 5)의 파라메터 포멧에 값을 설정하여 SIDIS에 전달한다.

<표 6> 사용자 속성 값 응답 포멧

| 파라메터 | 값 |
|---|---|
| openid.sidis.mode | response_attribute_value |
| openid.sidis.value.<alias> | 요청된 사용자 속성 값들 |
| openid.sidis.request_count | 요청된 사용자 속성 갯수 |
| openid.sidis.response_count | 응답된 사용자 속성 갯수 |

각각의 사용자 속성들은 설정된 정책에 반영하여 "meta" 테그를 이용한 redirect 방식으로 최종적으로 웹 사이트에 정보를 제공하게 된다. (그림 5)는 최종적으로 웹 사이트에 제공되어지는 사용자의 속성 정보를 사용자가 확인할 수 있게 하여, 사용자가 불필요한 정보의 유출이 있을 경우 이를 취소할 수 있도록 하는 인터페이스를 제공한다. 이같은 방법을 통해 실제 "사용자 중심"의 Identity 관리 기능을 강화할 수 있다.



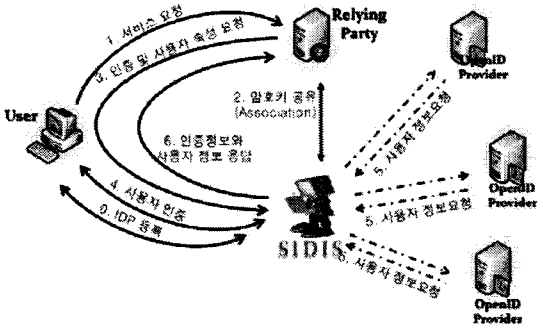

(그림 5) 사용자 정보 전달 제어 인터페이스

## 4. 시스템 분석 및 향후 과제

웹에서 제공되는 서비스들이 점점 진화하고 발전함에 따라 Identity관리 기술도 점점 다양하고 고도화된 기술들을 필요로 하고 있다. 초기의 Identity 관리 기술이 제한된 도메인 내에서 인증, SSO 및 인가 기술에 초점을 맞추어 기술이 개발된 반면 현재의 Identity 관리 기술은 사용자를 중심으로 Identity 정보의 공유에 보다 무게를 두고 개발되며 정보의 공유시 발생할 수 있는 프라이버시 문제에도 많은 연구를 하고 있다.

본 논문에서는, 사용자 중심의 Identity 관리 시스템의 단점을 보완하여 보다 편리하고 사용자가 자신의 정보의 흐름을 관리 할 수 있는 서비스를 구현하여 시험하였다.

여기서 제시된 SIDIS는 YADIS에서 제공하는 단순한 OpenID Provider를 지정하는 서비스 대신, 웹 서비스에서 요구하는 사용자의 정보에 따라 자동으로 사용자 정보 제공자를 선택하여 질의함으로서, 분산되어 있는 사용자의 정보를 통합하여 제공한다. 동시에 제공되는 사용자의 속성당 정책을 설정하여 사용자가 원하는 정보만




(그림 6) SIDIS의 전체 구조

이 제공될 수 있게 하며, 실시간으로 제공되는 사용자의 정보를 확인하여 보다 향상된 사용자 중심 Identity 관리 시스템을 제공하였다.

앞으로 향후 연구에서는 Identity의 온톨로지 기술로써, 다양한 Identity 관리 시스템에서 인식할 수 있는 Identity에 대한 기술 연구와 상이하게 표현된 Identity 간의 일치성을 증명할 수 있는 추론엔진에 대한 연구를 진행하며, SIDIS와 OpenID 제공자간의 인증 메카니즘을 계속 연구할 것이다.

## 참 고 문 헌

[1] 한국전자통신연구원 디지털Identity보안연구팀, "인터넷 Identity 관리 서비스 2006년도 기술 백서", 2006.
[2] 한국정보보호진흥원, "홈페이지 개인정보 노출원인과 대응방법", p.45, 2007. 6.
[3] Liberty Alliance Project, "Liberty ID-WSF Web Services Framework Overview", 2003.
[4] Liberty Alliance Project, "Liberty Alliance: Introduction to the Liverty Alliance Identity Architecture", 2003
[5] OpenID, http://openid.net
[6] LID, http://lid.netmesh.org
[7] Joaquin Miller, "Yadis Specification Version 1.0", http://yadis.org/papers/yadis-v1.0.pdf
[8] Microsoft, "Microsoft .NET Passport", 2004