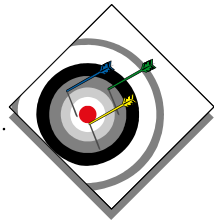


포/커/스

오픈소스 ID 관리 프로젝트 동향

김승현* 진승현**

정보 기술이 발전함에 따라 디지털 정보로 개인의 특성을 표현할 수 있는 방법들이 늘어나고 있다. 하지만 기존의 방법으로는 개인들을 제대로 식별하기 어려워지고 있다. 이를 해결하기 위해 여러가지 ID 관리 기술들이 등장하였지만, 오히려 다양한 ID 관리 기술들로 인해 상호운용성 문제가 우려된다. 오픈소스 형태의 ID 관리 프로젝트는 기존의 ID 문제뿐만 아니라 ID 관리 기술들의 상호운용성까지 해결해 주는 효과적인 방안이다. 최근 여러 업체들이 오픈소스 프로젝트에 주도적으로 참여하고 있으며, 대표적인 오픈소스 ID 관리 프로젝트로는 OpenSSO, Shibboleth, JAX-WS, OSIS, Higgins, Bandit, Heraldry 가 존재한다. 본 고에서는 이들 프로젝트를 살펴봄으로써 오픈소스 ID 관리 기술의 현황과 장단점에 대해 알아 보았다. ▧

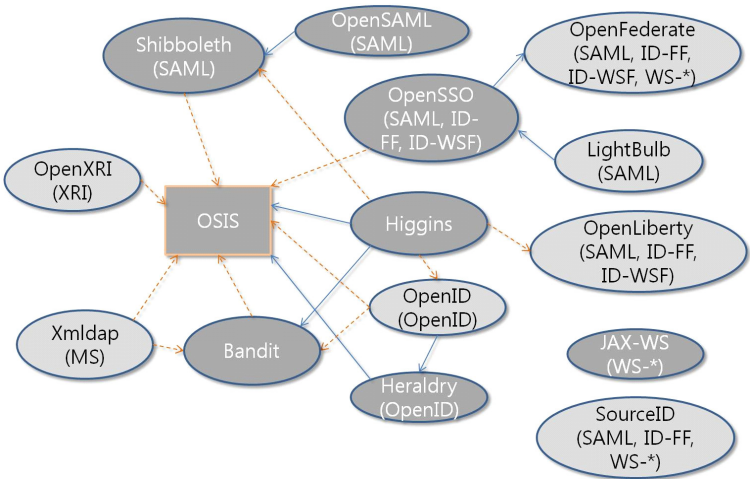
목 차

* ETRI 디지털 ID 보안연구팀/연구원
** ETRI 디지털 ID 보안연구팀/팀장

Ⅰ. 서 론

실생활에서 적용되는 거의 모든 활동을 인터넷 환경에서 제공받을 수 있는 시대가 되었다. 그러나 하나의 신분만으로 활동할 수 있는 현실과는 달리, 인터넷 환경에서는 여러 가지 ID(Identity)를 통해 사이트마다 다른 신분을 사용하게 된다. 이렇게 다양한 ID 는 각각의 사이트에서 개별적으로 관리되며, 본인의 정보는 인터넷 상에 산재되어 유지되기 때문에 보안 및 프라이버시에 취약하다는 문제점을 가지고 있다.

ID 관리 기술은 이러한 문제를 해결하기 위해서 등장한 기술로서, 인증·인가·접근 제어에서부터 감사 및 법률적 규제까지 다양한 분야를 다루고 있다. 관련 기술로는 인터넷 레벨에서 ID 를 관리하기 위해 1999 년 등장한 Microsoft 의 .NET Passport 서비



(그림 1) 오픈소스 ID 관리 프로젝트 동향

스를 시작으로 하여 WS-I 의 WS-* 표준, Liberty Alliance 의 ID-FF 및 ID-WSF 표준, OASIS 의 SAML 표준과 같은 기술들이 등장하였다. 최근에는 URL 을 ID 로 사용하는 대표적인 기술인 OpenID, Microsoft 의 사용자 중심 ID 관리 기술인 CardSpace 등이 주목을 받고 있다.

(그림 1)은 오픈소스로 운영되는 ID 관리 프로젝트의 목록과 이들 프로젝트들의 연관 관계를 보여준다. 각 항목은 오픈소스 ID 관리 프로젝트를 가리키는데, 타원형은 개발 프로젝트를 의미하고 사각형은 관리 프로젝트를 의미한다. 단, 항목의 크기에는 의미를 부여하지 않았다. 괄호로 표시된 부분은 해당 프로젝트가 중점적으로 지원하는 표준 기술을 의미한다. 화살표는 프로젝트의 종속 관계를 나타내는데, 실선은 이미 해당 프로젝트를 지원하는 관계를 의미하고 점선은 지원 예정인 관계를 의미한다.

ID 관리 프로젝트가 오픈소스로 진행하는 것은 다음과 같은 문제를 해결하기 위해서다.

- 인터넷 상의 ID 문제는 하나의 업체 또는 표준으로 해결할 수 있는 문제가 아님
- 고객들이 ID 관리 시스템들 간의 호환성과 구축의 용이성을 요구함
- 업계의(de facto) 표준화를 이루기 위해서는 광범위한 분야에 적용되어야 함
- 다양한 ID 관리 표준들이 존재하기 때문에 표준화 만으로는 업계의 동의를 얻기 어려움
- 표준을 준용하더라도 상호운용성 시험을 거쳐야지만 호환성을 보장할 수 있음

ID 문제는 여러 업체 및 조직이 공동으로 협력해야지만 해결할 수 있는 것으로 인식되고 있으며, 광범위한 분야에 먼저 적용될 수 있으면 업계 상의 표준으로 인정받기 때문에 오픈소스로 개발하는 것이 유리하다. 따라서 (그림 1)과 같이 여러 프로젝트가 오픈소스로 진행되고 있으며,

본 고에서는 대표적인 오픈소스 프로젝트로 OpenSSO, Shibboleth, JAX-WS, OSIS, Higgins, Bandit, Heraldry 를 소개한다.

II. 오픈소스 ID 관리 프로젝트

1. OpenSSO

OpenSSO[1] 프로젝트는 Sun 의 ID 관련 제품의 기반 기술을 담당하는 웹 SSO(Single Sign-On) 프로젝트로 2006 년 8 월 17 일에 릴리즈되었다. 이 프로젝트는 어떤 웹 애플리케이션에서도 사용 가능한 인증, 세션 로깅 서비스를 통해 ID 기반의 인증과 SSO 기능을 제공하는 서비스 인프라 역할을 수행한다.

OpenSSO 프로젝트가 제공하는 서비스 목록으로는 인증, 정책(인가) 관리, SAML, Federated Identity, 세션 관리, 로그 관리가 있다.

OpenSSO 의 하부 프로젝트로는 OpenFederation[2]과 LightBulb[3]가 있다. OpenFederation 은 ID Federation 과 웹 서비스 프레임워크를 개발하는 프로젝트로서 OASIS 의 SAML, Liberty Alliance 의 ID-FF, ID-WSF 표준을 지원한다. 그리고 LightBulb 는 SAML2.0 표준을 PHP 언어로 구현하는 프로젝트로서, PHP 에서 Java 클래스를 호출하는 Bridge 방식으로 개발되어 있다.

현재 OpenSSO 프로젝트는 Sun 의 최신 ID 관리 기술 제품인 Java System Access Manager 7.1 의 코어 ID 인프라로 포함되어 있으며, LightBulb 는 2006 년 10 월에 릴리즈한 이후로 OpenSSO 의 하부 프로젝트로 이동한 상태이다.

OpenSSO 와 하부 프로젝트는 Common Development and Distribution License(CDDL)[4] 라이선스를 가지고 있다.

2. Shibboleth

Shibboleth[5] 프로젝트는 미국의 교육 및 학술 분야를 위한 Internet2[6] 미들웨어 아키텍처 프로젝트로, 웹 기반 리소스를 위한 인증 및 접근 제어 시스템을 개발한다. Shibboleth 프로젝트의 구체적인 목적은 다음과 같다.

- OASIS SAML v1.1 표준을 완벽히 구현하여 제공

- OpenSAML 을 기반으로 하여 Federated SSO 와 Attribute Exchange 기능을 수행하기 위한 프레임워크 제공
- 사용자가 직접 Attribute 공유를 제어할 수 있는 확장된 프라이버시 기능 제공

타 ID 관리 프로젝트와 비교하여 Shibboleth 프로젝트는 몇 가지 중요한 특징을 가진다.

<표 1> Shibboleth 프로젝트의 특징

특징	설명
연방화된 관리	- IDP 는 SP 에게 속성 확인정보(assertion) 제공 - SP 별로 신뢰 레벨을 설정하여 관리
속성 정보에 기반한 접근 제어	- ID 가 아닌 속성 정보로 대부분의 접근 제어 처리
적극적인 프라이버시 관리	- 웹 기반 인터페이스로 속성 정보의 공개 제어
표준 기반	- SAML 사용
다양하고 확장 가능한 신뢰 정책 집합을 제공하는 프레임워크	- 일반적인 정책 집합 명세 - 신뢰 프레임워크를 통해 상황에 따른 정책 적용이 유연하게 제공되도록 함
속성 정보의 표준 어휘 집합	- 속성 정보의 구분을 위한 표준 집합 정의

Shibboleth 프로젝트에서 핵심이 되는 하부 프로젝트로 Java 와 C++ 언어로 SAML 스펙을 구현하는 OpenSAML[7] 프로젝트가 있다. 이 프로젝트는 OpenSAML 1.1 버전을 작성 완료하였고, 현재 2.0 버전을 작성중이다. 1.1 버전은 Apache 의 XML 프로젝트인 Xerces 와 XML-Security 를 기반으로 만들어졌고, 2.0 버전은 기존의 SAML 1.x 버전을 지원하면서 SAML 2.0 스펙을 개발중이다.

Shibboleth 프로젝트는 현재 1.3 버전이 릴리즈되었으며, 2.0/2.1 버전에서는 OpenSAML 2.0 과의 통합을 목표로 한다. OpenSAML 2.0 은 현재 50% 정도 개발되었으며, 2007 년 내에 완료될 예정이다.

Shibboleth 와 OpenSAML 프로젝트는 Apache 2.0 License[8] 라이선스를 가지고 있다.

3. JAX-WS

JAX-WS[9] 프로젝트는 Sun 의 GlassFish[10] 프로젝트를 구성하는 하부 프로젝트로서, XML 웹 서비스를 위한 Java 용 API(JAX-WS) 스펙 구현을 목표로 한다. 이 기술은 웹 서비스 간의 상호운용을 가능하게 하는 기반 기술을 제공하여 Microsoft 의 통신 기술과의 상호운영을 가능하게 만들어 준다. JAX-WS 는 다음의 표준을 지원한다.

- JAX-WS 2.0/2.1
- JAXB 2.1
- WS-I Basic Profile 1.1
- WS-I Attachments Profile 1.0
- WS-I Simple SOAP Binding Profile 1.0
- WS-Addressing 1.0 : 코어, SOAP 바인딩, WSDL 바인딩

JAX-WS 의 하부 프로젝트로는 'Project Tango'로 명명된 WSIT[11]가 있다. WSIT 프로젝트는 WS-* 스펙을 구현하는 목표를 가지고 있으며, 구체적으로 <표 2>의 기능 및 표준을 지원한다.

<표 2> WSIT 프로젝트의 기능

기능	설명 및 지원 표준
초기화(Bootstrapping) 통신	- WCF(Windows Communications Foundataion) 기반 서비스로부터 서비스 WSDL을 획득하기 위한 서비스 - WS-Metadata Exchange, WS-Metadata Exchange WSDL, WS-Transfer
최적화(Optimizing) 통신	- 최적화된 바이너리 인코딩을 통해 메시지 크기를 줄이고 보안 통신을 위한 초기 설정을 수행하는 서비스 - WS-Secure Conversation, WS-Policy, WS-Policy Attachment
신뢰성(Reliability)	- 메시지 손실 또는 순서오류 문제로 인한 전송 실패를 복구하는 서비스 - WS-Reliable Messaging, WS-Reliable Messaging Policy
단일(Atomic) 트랜잭션	- 한 트랜잭션 경계 내에 있는 모든 연산이 성공적으로 수행되었음을 보장하는 서비스 - WS-Atomic Transaction, WS-Coordination
보안 통신	- 메시지 레벨에서의 보안 기능과 보안 토큰 생성 기능을 제공하는 서비스 - WS-Security, WS-Security Policy, WS-Trust

JAX-WS 프로젝트는 2007 년 2 월에 JAXWS 2.1 버전을 릴리즈하였다. WSIT 프로젝트는 2007 년 2 월에 베타버전을 공개하였으나, WS-MetadataExchange, WS-Security, WS-Atomic Transaction, WS-SecurityPolicy 표준의 지원이 부족한 상태이다.

JAX-WS 와 하부 프로젝트는 Common Development and Distribution License(CDDL)[4] 라이선스를 가지고 있다.

4. OSIS

OSIS[12] 프로젝트는 2006 년 6 월에 개최된 Identity Mashup 컨퍼런스에서 발족된 프로젝트로서 Microsoft, Novell, IBM, SXIP, XRI, Verisign 과 같은 여러 업체가 참여하고 있다. 프로젝트 설립 초기에는 Microsoft 의 CardSpace 제품에 대한 오픈소스 버전을 작성한다는 목표를

가졌으나, 지금은 그 범위를 확장하여 오픈소스 ID 관리 프로젝트들을 체계적으로 관리한다는 목표를 설정하였다. OSIS 프로젝트는 오픈소스 프로젝트들의 업무가 중복되지 않도록 조율하고, 공통 소프트웨어 인터페이스를 작성하며, 구현물 간의 상호운용성을 제공하려는 목표를 가진다. 이를 위해 <표 3>과 같은 위원회를 구성하여 운영하고 있다.

<표 3> OSIS 프로젝트의 위원회

구분	내용
운영 위원회	- 오픈소스 프로젝트와 기업의 대표 결정권자들로 구성 - OSIS 프로젝트의 운영 방안 논의
설계 위원회	- 각 오픈소스 ID 관리 프로젝트의 담당자 1 인으로 구성 - OSIS 아키텍처 설계에 대한 논의
라이선스와 배포 위원회	- 기업의 대표와 OSIS 소프트웨어의 배포자들로 구성 - 오픈소스의 라이선스 및 배포 방안에 대한 논의

OSIS 는 OpenID 로 대표되는 URL 기반의 시스템과, CardSpace 및 Higgins 로 대표되는 카드(토큰) 기반의 ID 시스템 간의 상호연동을 제공하려 한다. 현재 Bandit, Heraldry, Higgins, OpenSSO, OpenXRI, Shibboleth, Xmldap 프로젝트가 OSIS 에 참여하고 있으며, OSIS 는 Apache 의 Heraldry 프로젝트가 서버 역할을 담당하고 Eclipse 의 Higgins 프로젝트가 클라이언트 역할을 담당하는 아키텍처를 제안한 상태이다. 이 아키텍처는 Microsoft 의 CardSpace 와도 호환되며, 향후에는 OpenID 와 SAML 또한 지원할 계획이다.

5. Higgins

Higgins[13] 프로젝트는 Novell 과 IBM 에 의해 시작되었으며 Eclipse 재단이 주관하는 프로젝트이다. Higgins 는 재사용이 가능하고, 플랫폼과 ID 프로토콜에 무관하며, 간편하게 프라이버시와 ID 정보를 제어할 수 있는 소프트웨어 프로토콜을 제공하는 것을 목표로 한다. 이를 위해 서로 다른 운영체제나 하드웨어를 사용하는 시스템 간에 ID ・프로파일・관계(relationship) 정보를 가상으로 통합시키는 기능, 로컬 시스템 또는 원격 시스템에 구애받지 않고 호출할 수 있는 ID 에이전트, 브라우저 확장 기능이 개발된다. Higgins 는 <표 4>와 같은 컴포넌트들로 구성된다.

Higgins 는 2006 년 중반에 초기 버전이 릴리즈된 이후로, 스펙 수정 작업과 여러 종류의 HBX 데모 개발 작업이 이루어졌다. 현재 CardSpace 와 OpenID 프로토콜 지원 작업을 완료하였으며, 차기 버전에는 SAML/Liberty 프로토콜을 지원할 계획이다.

Higgins 프로젝트는 Eclipse Public License(EPL)[14] 라이선스를 가지고 있다.

<표 4> Higgins 프로젝트의 컴포넌트

컴포넌트	내용
Higgins Browser Extension(HBX)	브라우저와 연계하여 사용자가 직관적으로 ID 정보를 카드 형태로 선택할 수 있도록 함
I-Card Selector Service(ISS)	Relying Party 가 요구하는 보안 정책을 만족하는 I-Card 를 선별하며, 이 과정에서 정책 엔진이 사용됨
I-Card Provider	I-Card 객체의 관리를 담당하며, 지원하는 I-Card 는 CardSpace 형식과 URI 형식, Managed 타입과 Personal 타입이 존재함
Identity Attribute Service(IdAS)	특정 컨텍스트(context)를 조건별로 검색하며, 컨텍스트의 URI 로 ID 속성 정보의 생성, 접근, 발행, 삭제 기능을 수행함
Context Provider	ID 속성들을 컨텍스트(context) 형태로 암호화하여 저장하며, 안전하고 일관성 있는 방식으로 컨텍스트를 제공함

6. Bandit

Bandit[15] 프로젝트는 Novell 이 주관하는 프로젝트로 인증, 인가, 감사(audit)와 같은 ID 서비스를 loosely-coupled 구조의 컴포넌트로 제공한다. Bandit 은 다양한 인증 기법을 사용할 수 있는 인터페이스뿐만 아니라 사용자 중심의 크리덴셜 관리 기능을 제공하며, 운영체제・플랫폼・설치된 보안 솔루션 기술에 구애받지 않고 일관성 있는 환경을 만들어준다.

Bandit 프로젝트는 Higgins 프로젝트의 공통 ID 컴포넌트와 연계하여 ID 인프라를 제공한다. Bandit 은 <표 5>와 같은 컴포넌트들로 구성되어 있다.

Bandit 은 2007 년 2 월까지 오픈소스 프로젝트로 CardSpace 와 Liberty Alliance 의 스펙을 지원하였으며, 2007 년 3 월에는 다양한 플랫폼 환경에서 구동되는 Identity Selector 을 개발하고 OpenID 스펙을 지원하였다. 향후에는 WS-*와 Liberty Federation 표준을 지원하며, Pamela Project, xmldap.org, OpenID, OSIS, Identity Commons 프로젝트와 연계할 계획을 가지고 있다.

Bandit 프로젝트는 GNU Lesser General Public License(LGPL)[16] 라이선스를 가지고 있다.

<표 5> Bandit 프로젝트의 컴포넌트

컴포넌트	내용
CASA (Common Authentication Service Adapter)	SSO(Single Sign-On) 기능과 여러 크리덴셜의 단일 관리 기능을 제공하며, 향후에는 정책 엔진 모듈과 저장소 동기화 기능을 제공할 계획임
Audit Record Framework (OpenXDAS)	단일 감사 레코드 포맷을 정의하고, 표준화된 감사 이벤트 분류법을 제공함
Role Engine	RBAC 과 XACML 로 개발되어 있으며, 현재 세션에서 사용할 수 있는 여러 종류의 역할 중에서 해당 목적에 최적화된 역할을 선택하여 활용할 수 있음

7. Heraldry

Heraldry[17] 프로젝트는 Apache 재단에서 관리하는 인큐베이터 프로젝트로 사용자 중심의 ID 공간을 제공하기 위한 기술을 개발한다는 목적을 가진다. 현재는 OpenID 표준을 준용한 라이브러리와 애플리케이션 개발에 초점을 맞추고 있다. Heraldry 는 OpenID 와 Yadis 표준을 구현한 라이브러리를 상용화 수준으로 제공할 계획이며, Microsoft 의 CardSpace 에서도 활용될 수 있는 라이브러리를 개발할 예정이다.

Heraldry 프로젝트는 다음과 같은 컴포넌트로 구성된다.

<표 6> Heraldry 프로젝트의 컴포넌트

컴포넌트	내용
Yadis	URI/XRI 기반의 서비스 검색 기능을 제공함
OpenID	웹 기반의 SSO 와 Profile 데이터 교환 서비스를 제공함

Heraldry 는 JanRain 이 운영하는 OpenIDEnabled[18]의 관련 자료와 VeriSign 에서 OpenID 서버를 구현한 PIP[19]의 소스를 제공받았다. 하지만 JanRain 의 프로젝트 참여 방식이 문제가 되어, Heraldry 프로젝트는 JanRain 을 멤버에서 제외하고 새로운 멤버를 모집하여 개발을 진행하고 있다.

Heraldry 프로젝트는 Apache 2.0 License[8] 라이선스를 가지고 있다.

III. 오픈소스 ID 관리 프로젝트의 장단점

1. 장점

오픈소스 ID 관리 프로젝트를 통해 얻을 수 있는 장점으로 크게 3 가지를 들 수 있다. 첫번째는 관련 기업들이 ID 문제에 공동으로 대처할 수 있다는 점이다. 지금까지 업체마다 독자적인 방식으로 중복 개발되었던 코드를 오픈소스로 대체함으로써 재사용성을 높일 수 있으며, 공개된 소스의 검증을 통해 안정성을 확인할 수 있게 된다. 또한 외부의 전문가를 이용할 수 있기 때문에 저렴하고 수준 높은 코드를 기대할 수 있다.

두번째로 오픈소스를 통한 호환성과 구축의 용이성을 기대할 수 있다. 기존에는 업체들이 동일한 표준을 준용하더라도, 상호운용이 보장되지 않기 때문에 부가적인 연동 작업이 요구되었다. 하지만 오픈소스로 개발된 동일한 모듈을 사용한다면 상호운용성을 쉽게 보장할 수 있다. 또한 누구나 오픈소스 프로젝트를 이용하여 시스템을 구축할 수 있기 때문에 해당 기술이 시장에 대

규모로 적용될 수 있다는 이점을 가진다.

세번째는 오픈소스를 통해 인터넷 상의 ID 계층을 형성함으로써, 개발자가 서비스에 집중할 수 있다는 점이다. 인터넷 상에 통용될 수 있는 인증·인가·신뢰 기능을 제공하는 프레임워크는 안전한 인터넷 환경을 구축해준다. 개발자는 자신들이 제공하려는 서비스에 좀 더 집중할 수 있으며, ID 계층을 통해 사용자의 인증 정보뿐만 아니라 신뢰 정보까지도 안전하게 교환할 수 있을 것이다.

2. 단점

오픈소스 ID 관리 프로젝트의 활동에서 우려되는 요소는 크게 2 가지를 들 수 있다. 첫번째는 오픈소스 ID 관리 프로젝트를 추진하는 주체들의 이해관계이다. 일반적인 오픈소스 프로젝트와 마찬가지로 ID 관리 프로젝트의 진행 여부가 구성원들의 의지로 이루어지기 때문에 프로젝트의 진행 여부가 불투명하다. 또한 관련 기업이 오픈소스 프로젝트에 참여한 경우에도, 자사의 핵심 기술 및 인력을 오픈소스 프로젝트에 적극적으로 투입할지에 대한 여부는 강제할 수 없다는 문제가 있다. 일례로 Heraldry 프로젝트의 JanRain 이 자신들의 기여도에 걸맞는 권한을 요청해서 문제가 된 경우를 들 수 있다.

두번째는 지적재산권 및 저작권 문제이다. 특허를 비롯한 지적재산권이 걸린 기술은 오픈소스로 제공할 경우에 문제의 소지가 있다. Microsoft 와 IBM 의 WS-* 관련 표준 또한 최근까지 특허로 보호받았기 때문에 오픈소스화 되기 어려웠다. 또한 각각의 오픈소스 프로젝트마다 다양한 종류의 라이선스를 사용하기 때문에 오픈소스를 활용하려는 사용자가 직접 해당 라이선스의 권한 범위를 제대로 파악해야 하는 불편함이 있다.

IV. 결 론

본 고에서는 오픈소스로 진행되고 있는 ID 관리 프로젝트인 OpenSSO, Shibboleth, JAX-WS, OSIS, Higgins, Bandit, Heraldry 에 대하여 간략하게 소개하였다. 서두에서 언급하였듯이 인터넷 상의 ID 문제는 하나의 업체 또는 하나의 표준만으로 해결할 수 있는 문제가 아니다. 여러 ID 관리 기술과 시스템들 간의 상호호환 기능을 제공할 수 있는 방안이 가장 이상적이며, 이를 위해 오픈소스 프로젝트들이 등장하고 있는 실정이다. 이미 ID 관리 분야에서 주도적인 위치에 있는 Microsoft, Sun, Novell, IBM 과 같은 업체뿐만 아니라, Liberty Allliance 나 OSIS 의 XRI/XDI 와 같은 조직 또한 오픈소스에 적극 동참하고 있다.

오픈소스 ID 관리 프로젝트의 도입을 위해 가장 먼저 해결해야 할 요소는 지적재산권 문제이다. 오픈소스로 구현하려는 기술 자체가 지적재산권으로 보호받거나, 오픈소스를 사용하고 싶지만 라이선스가 맞지 않아 사용을 포기해야 할 경우가 생기기 쉽다. 하지만 2006 년 11 월, Microsoft 가 WS-* 관련 표준의 오픈소스화를 공표[20]한 이후로 WS-*를 개발하거나 지원을 준비중인 오픈소스 프로젝트가 다수 등장하고 있다. 또한 OSIS 가 오픈소스 ID 관리 프로젝트를 관리하면서 다양한 종류의 라이선스 또한 간단하게 정리해 줄 것으로 기대된다.

(그림 1)에서 볼 수 있듯이 여러 오픈소스 ID 관리 프로젝트들은 긴밀한 연관관계를 통해 시너지 효과를 기대하고 있으며, OSIS 와 같은 단체를 통해 프로젝트들 간의 업무 조율을 도모하고 있는 실정이다. 이에 따라 오픈소스 ID 관리 프로젝트들간에 긴밀한 공조가 이루어질 것이며 관련 시장 또한 적극적으로 오픈소스를 도입할 것이다.

<참 고 문 헌>

[1] OpenSSO, https://opensso.dev.java.net/
[2] OpenFederation, https://opensso.dev.java.net/
[3] LightBulb, https://lightbulb.dev.java.net/
[4] Common Development and Distribution License
[5] Shibboleth, https://shibboleth.internet2.edu/
[6] Internet 2 Project, http://www.internet2.edu/
[7] OpenSAML, http://www.opensaml.org/
[8] Apache 2.0 License, http://www.apache.org/licenses/LICENSE-2.0.html
[9] JAX-WS(Java API for XML Web Services), https://jax-ws.dev.java.net/
[10] GlassFish Project, http://glassfish.dev.java.net/
[11] WSIT(Web Services Interoperability Technologies), https://wsit.dev.java.net/
[12] OSIS(Open Source Identity System), http://osis.netmesh.org/wiki/Main_Page
[13] Higgins, http://www.eclipse.org/higgins/
[14] Eclipse Public License(EPL), http://www.eclipse.org/legal/epl-v10.html
[15] Bandit, http://www.bandit-project.org/
[16] GNU Lesser General Public License(LGPL), http://www.gnu.org/licenses/lgpl.html
[17] Heraldry, http://incubator.apache.org/projects/heraldry.html
[18] OpenIDEnabled, http://www.openidenabled.com/
[19] PIP(Personal Identity Provider), http://pip.verisignlabs.com/
[20] Microsoft Open Specification Promise, http://www.microsoft.com/interop/osp/

* 본 내용은 필자의 주관적인 의견이며 IITA 의 공식적인 입장이 아님을 밝힙니다.