TTA Standard

정보통신단체표준(국문표준)

TTAS.IT-X1141_3

개정일: 2006 년 12 월 27 일

SAML 2.0 프로파일

Profiles for SAML 2.0

한국정보통신기술협회
Telecommunications Technology Association

정보통신단체표준(국문표준)

TTAS.IT-X1141_3 개정일: 2006 년 12 월 27 일

SAML 2.0 프로파일

Profiles for SAML 2.0

한국정보통신기술협회
Telecommunications Technology Association

서 문

1. 표준의 목적

SAML(Security Assertion Markup Language) 2.0 은 분산된 환경에서 인증, 인가 및 속성 정보를 통신하기 위한 XML-기반 프레임워크이다. 이름에서 나타나듯이, SAML 은 비즈니스 엔티티들이 어떤 주체의 신원, 속성 그리고 권한부여에 대한 주장을 파트너 회사 또는 다른 엔터프라이즈 응용 등과 같은 다른 엔티티들에게 주장하는 것을 허용한다. 이 표준은 통신 프로토콜과 프레임워크에서 SAML 주장과 요청-응답 메시지들의 사용을 위한 프로파일들을 정의하며 또한 SAML 속성 값 문법과 명명 규정들을 정의한다.

이 표준은 ITU-T X.1141 “Security Assertion Markup Language (SAML 2.0)”을 근거로 한 국내 표준으로 원문의 다음 내용을 포함하고 있다.

CL 1. 범위
CL 2. 참고문헌
CL 3. 용어정의
CL 4. 약어
CL 5. 관례
CL 7. 공통 데이터 타입
CL 10. SAML 프로파일

2. 주요 내용 요약

이 표준은 처음에 프로파일의 개념에 대하여 기술한다. 그 다음, 이 표준은 추가적인 프로파일들을 명기하는 것에 대한 가이드라인과 확인 방법 식별자들을 기술한다. 또한, 이 표준은 SSO 프로파일들, Artifact Resolution 프로파일, Assertion Query/Request 프로파일, Name Identifier Mapping 프로파일 그리고 SAML 속성 프로파일들을 정의한다.

3. 표준 적용 산업 분야 및 산업에 미치는 영향

본 표준은 웹 싱글사인온, 속성-정보 기반 인가와 웹 서비스 보호에서 사용될 수 있다. 따라서, 본 표준은 ID 관리 분야와 웹 서비스 정보보호 분야에 직접적으로 적용되며, 정보보호 산업의 핵심 요소로 활용될 수 있다. 또한, ID 연계의 핵심 기술을

제공함으로써, 기업 간 협업을 용이하게 함으로써 새로운 서비스를 창출하고 시장을 활성화할 수 있다.

4. 참조 표준(권고)

4.1. 국외 표준(권고)

- ITU-T X.1141, 'Security Assertion Markup Language (SAML 2.0)', 2006.06.

4.2. 국내 표준

- TTAS.OT-10.0042, 'SAML 바인딩과 프로파일', 2005.12.

5. 참조 표준(권고)과의 비교

5.1. 참조 표준(권고)과의 관련성

본 ITU-T X.1141, "Security Assertion Markup Language (SAML 2.0),"을 근거로 한 국내표준임. TTAS.OT-10.0041 는 OASIS 의 SAMLv1.0 을 기준으로 개발된 표준이였으나 현재 SAMLv2.0 표준이 ITU-T 표준으로 개발되어 이를 개정함.

5.2. 참조한 표준(권고)과 본 표준의 비교표

상기 국제 권고에 대한 추가사항은 없으며 장 구성은 다음과 같음.

ITU-T X.1141	본 표준	비고
1. 범위	1.1. 범위	동일(번역)
2. 참고문헌	1.2. 참고문헌	동일(번역)
3. 용어정의	1.3. 용어정의	동일(번역)
4. 약어	1.4. 약어	동일(번역)
5. 관례	1.5. 관례	동일(번역)
7. 공통 데이터 타입	1.6. 공통 데이터 타입	동일(번역)
11. SAML 프로파일	2~5. SAML 프로파일	동일(번역)

6. 지식 재산권 관련 사항

본 표준의 ‘지식 재산권 확약서’ 제출 현황은 TTA 웹사이트에서 확인할 수 있다.

※본 표준을 이용하는 자는 이용함에 있어 지식 재산권이 포함되어 있을 수 있으므로, 확인 후 이용한다.

※본 표준과 관련하여 접수된 확약서 이외에도 지식 재산권이 존재할 수 있다.

7. 시험 인증 관련 사항

7.1. 시험 인증 대상 여부

- 해당 사항 없음.

7.2. 시험 표준 제정 현황

- 해당 사항 없음.

8. 표준의 이력 정보

8.1. 표준의 이력

판수	제정·개정일	제정·개정 내역
제 1 판	2005.12.21.	제정 TTAS.IT-X1141
제 2 판	2006.12.27.	개정 TTAS.IT-X1141_3

8.2. 주요 개정 사항

- 해당 사항 없음.

Preface

1. Purpose of Standard

SAML(Security Assertion Markup Language) is an XML-based framework for communicating user authentication, entitlement, and attribute information among disparate Web access management and security products. As its name suggests, SAML allows business entities to make assertions regarding the identity, attributes, and entitlements of a subject (an entity that is often a human user) to other entities, such as a partner company or another enterprise application. This standard defines profiles for the use of SAML assertions and request-response messages in communications protocols and frameworks, as well as profiles for SAML attribute value syntax and naming conventions.

This standard is a domestic standard based on ITU-T X.1141 “Security Assertion Markup Language (SAML 2.0)” and contains the following contents of the original standard.

CL 1. Scope
CL 2. References
CL 3. Definitions
CL 4. Abbreviations
CL 5. Conventions
CL 7. Common data types
CL 11. Profiles for SAML

2. Summary of Contents

The standard specifies the concept of profile at first. Then, it specifies guidelines for specifying additional profiles and confirmation method identifiers. It also defines SSO Profiles, Artifact Resolution Profile, Assertion Query/Request Profile, Name Identifier Mapping Profile and SAML Attribute Profiles.

3. Applicable Fields of Industry and its Effect

This standard can be used in as web single-sign on, attribute information based authorization and web service security. Therefore, it is directly applicable to security areas such as ID Management and web service security. It is also applicable to other information security industry as essential component. In addition, it provides essential technology for ID federation, which makes companies' collaboration easy and so creates new service and revitalize IT market.

4. Reference Standards(Recommendations)

4.1. International Standards(Recommendations)

- ITU-T X.1141, "Security Assertion Markup Language (SAML 2.0)," 2006.06.

4.2. Domestic Standards

- TTAS.OT-10.0041, "SAML Assertions and Profile", 2005.12.

5. Relationship to Reference Standards(Recommendations)

5.1. Relationship of Reference Standards(Recommendations)

This standard is a domestic standard based on ITU-T X.1141, "Security Assertion Markup Language (SAML 2.0)". This standard updated the TTAS.OT-10.0041 based on OASIS standard as SAMLv1.0.

5.2. Differences between Reference Standard(Recommendation) and this Standard

This standard has no additional contents as to the international recommendations. The differences between the recommendation and this standard are as follows.

ITU-T X.1141	This Standard	
1. Scope	1.1. Scope	equaled(trans)
2. References	1.2. References	equaled(trans)

3. Definitions	1.3. Definitions	equaled(trans)
4. Abbreviations	1.4. Abbreviations	equaled(trans)
5. Conventions	1.5. Conventions	equaled(trans)
7. Common data types	1.6. Common data types	equaled(trans)
11. Profiles for SAML	2~5. Profiles for SAML	equaled(trans)

6. Statement of Intellectual Property Rights

IPRs related to the present document may have been declared to TTA. The information pertaining to these IPRs, if any, is available on the TTA Website.

No guarantee can be given as to the existence of other IPRs not referenced on the TTA website.

And, please make sure to check before applying the standard.

7. Statement of Testing and Certification

7.1. Object of Testing and Certification

- None

7.2. Standards of Testing and Certification

- None

8. History of Standard

8.1. Change History

Edition	Issued date	Outline
The 1st edition	2005.12.21.	Established TTAS.IT-X1141
The 2nd edition	2006.12.27.	Revised TTAS.IT-X1141_3

8.2. Revisions

- None

목 차

Contents

SAML 2.0 프로파일
(Profiles for SAML 2.0)

1. SAML 2.0 개요

1.1. 범위(Scope)

SAML 2.0 은 시스템 엔티티가 어떤 주체에 대하여 생성한 주장의 문법과 처리 규칙을 정의한다. 이와 같은 주장을 만들거나 또는 의지하기 위해, SAML 시스템 엔티티들은 주장 자체 또는 주장의 주체에 대한 내용을 통신하기 위해 다른 프로토콜을 사용할 수 있다. SAML 2.0 은 SAML 보장의 구조, 관련된 프로토콜 집합, 그리고 SAML 시스템을 관리하는데 관련된 처리 규칙들을 정의한다.

SAML 주장과 프로토콜 메시지들은 XML 로 인코딩되어 있으며, XML 네임스페이스를 사용한다. 이것들은 일반적으로 HTTP POST 또는 XML 로 인코딩된 SOAP 메시지와 같은 전송을 위한 다른 구조에 내장된다. SAML 2.0 은 또한 SAML 프로토콜 메시지들을 내장하고 전송하기 위한 프레임워크를 제공하는 SAML 바인딩을 명기한다. 더욱이, SAML 2.0 은 SAML 특징들을 사용할 때, 특정 사용예(use case)를 달성하고 상호운용성을 달성하기 위해, SAML 주장과 프로토콜을 어떻게 사용해야 하는지에 대한 기본 프로파일 집합을 제공한다.

SAML 2.0은 다음을 정의한다.

1. SAML 에 대한 적합성 요구사항;
2. SAML 주장과 프로토콜:
   - SAML 주장 스키마,
   - SAML 프로토콜 스키마;
3. SAML 바인딩;
4. SAML 프로파일:
   - SAML ECP 프로파일 스키마,
   - SAML X.500/LDAP 속성 프로파일 스키마,
   - SAML DCE PAC 속성 프로파일 스키마,
   - SAML XACML 속성 프로파일 스키마;
5. SAML 메타데이터;
6. SAML 메타데이터 스키마;
7. SAML 인증 문맥.

1.2. 참고 문헌

다음 권고안들과 다른 참조들은 SAML 2.0 에서 참조되는 것들이다. SAML 2.0 의 발간시에는 모두 유효한 상태이다. 모든 권고안들과 다른 참조들은 개정될 수 있으며, SAML 2.0 에 기반으로 하는 모든 사용자들은 아래 나열된 권고안들과 다른 참조들에 대하여 가장 최신 판을 적용할 수 있다. ITU 의 전기통신 표준국(Telecommunications Standardization Bureau)에서 현재 유효한 ITU-T 권고안들의 리스트를 유지한다. ITEF 는 최근에 폐지된 것들과 함께 RFC 리스트를 유지한다. W3C, Unicode Consortium 과 Liberty Alliance 도 가장 최신의 권고안들과 다른 문서들에 대한 리스트를 유지한다.

- ITU-T Recommendation X.660 (2004), Information technology - Open Systems Interconnection - Procedures for the operation of OSI Registration Authorities: General procedure. .
- ITU-T Recommendation X.667 (2004), Information technology - Open Systems Interconnection - Procedures for the operation of OSI Registration Authorities: Generation and Registration of Universally Unique Identifiers (UUIDs) and their Use as ASN.1 Object Identifier Components.
- ITU-T Recommendation X.680 (2002), Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation.
- ITU-T Recommendation X.800 (1991), Information processing systems - Open Systems Interconnection - Basic Reference Model - Part 2: Security Architecture.
- ITU-T Recommendation X.811 (1995), Security Frameworks for Open Systems: Authentication Framework.
- ITU-T Recommendation X.812 (1995), Security Frameworks for Open Systems: Access control framework.
- ITU-T Recommendation X.1142 (2006), Extensible Access Control Markup Language (XACML 2.0).
- IETF RFC 1034:1987, Domain Names - Concepts and Facilities, 1987.
- IETF RFC 1510:1993, The Kerberos Network Authentication Requestor (V5), 1993.
- IETF RFC 1750:1994, Randomness Recommendations for Security, 1994.
- IETF RFC 1951:1996, DEFLATE Compressed Data Format Specification Version 1.3, 1996.
- IETF RFC 1991:1996, PGP Message Exchange Formats, 1996.
- IETF RFC 2045:1996, Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message, 1996.
- IETF RFC 2119:1997, Key words for use in RFCs to Indicate Requirement Levels, 1997.
- IETF RFC 2246:1999, The TLS Protocol Version 1.0, 1999.

- IETF RFC 2253:1997, Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished, 1997.
- IETF RFC 2396:1998, Uniform Resource Identifiers (URI): Generic Syntax, 1998.
- IETF RFC 2535:1999, Domain Name System Security Extensions, 1999.
- IETF RFC 2616 :1999, Hypertext Transfer Protocol - HTTP/1.1, 1999.
- IETF RFC 2617:1999, HTTP Authentication: Basic and Digest Access Authentication, 1999.
- IETF RFC 2798:2000, Definition of the inetOrgPerson LDAP Object Class, 2000.
- IETF RFC 2828:2000, Internet Security Glossary, 2000.
- IETF RFC 2914:2000 , Congestion Control Principles, 2000.
- IETF RFC 2915:2000, The Naming Authority Pointer (NAPTR) DNS Resource Record, 2000.
- IETF RFC 2945:2000, The SRP Authentication and Key Exchange System, 2000.
- IETF RFC 2965:2000, HTTP State Management Mechanism, 2000.
- IETF RFC 3061:2001, A URN Namespace of Object Identifiers, 2001.
- IETF RFC 3075:2001, XML-Signature Syntax and Processing, 2001.
- IETF RFC 3513:2003, Internet Protocol Version 6 (IPv6) Addressing Architecture, 2003.
- IETF RFC 3023:2001, XML Media Types, 2001.
- IETF RFC 3377:2002, Lightweight Directory Access Protocol (v3): Technical Specification, 2002.
- IETF RFC 3403:2002, Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database, 2002.
- IETF RFC 3546:2003, Transport Layer Security (TLS) Extensions, 2003.
- IETF RFC 3923:2004, End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP), 2004.
- IETF RFC 4122:2005, A Universally Unique IDentifier (UUID) URN Namespace, 2005.
- Liberty Alliance POAS:2003, R. Aarts, Reverse HTTP Binding for SOAP Specification Version 1.0, Liberty Alliance Project, 2003.
- OASIS WSS:2006, WS-Security Core Specification 1.1, February, 2006.
- UNICODE-C, M. Davis, M. J. Dürst, Dürst. Unicode Normalization Forms. UNICODE Consortium, March 2001.
- W3C Canonicalization:2002, Exclusive XML Canonicalization Version 1.0, W3C Recommendation, Copyright © [2 July 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xml-exc-c14n/.
- W3C Character Model:2005, Character Model for the World Wide Web 1.0: Fundamentals, W3C Recommendation, Copyright © [15 February 2005] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2005/REC-charmod-20050215/.
- W3C Datatypes:2001, XML Schema Part 2: Data types, W3C Recommendation, Copyright ©

[2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/.

- W3C Encryption:2002, XML Encryption Syntax and Processing, W3C Recommendation, Copyright © [10 December 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/.
- W3C Web Services Glossary:2004, Web Services Glossary, W3C Note, Copyright © [11 February 2004] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/ws-gloss/.
- W3C HTML:1999, HTML 4.01 Specification, W3C Recommendation, Copyright © [24 December 1999] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/REC-html40/.
- W3C Namespaces:1999, Namespaces in XML, W3C Recommendation, Copyright © [14 January 1999] World Wide Web Consortium (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/REC-xml-names/.
- W3C Primer:2005, SOAP Version 1.2 Part 0: Primer, W3C Recommendation, Copyright © [24 June 2005] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2003/REC-soap12-part0-20030624/.
- W3C Signature:2002, XML Signature Syntax and Processing, W3C Recommendation, Copyright © [12 February 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xmldsigcore/.
- W3C Signature Schema:2001, XML Signature Schema, W3C Recommendation, Copyright © [1 March 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xmldsig-core/xmldsig-core-schema.xsd.
- W3C String:1998, Requirements for String Identity Matching and String Indexing, W3C Note, Copyright © [10 July 1998] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/WD-charreq.
- W3C SOAP:2000, Simple Object Access Protocol (SOAP) 1.1, W3C Note, Copyright © [08 May 2000] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University),

http://www.w3.org/TR/2000/NOTE-SOAP-20000508.

- W3C XHTML:2002, The Extensible HyperText Markup Language (Second Edition), W3C Recommendation, Copyright © [1 August 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xhtml1/.
- W3C XML 1.0:2004, Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation, Copyright © [4 February 2004] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/REC-xml/.
- W3C XML Schema Part 1:2001, XML Schema Part 1: Structures, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/.

주의 - SAML 2.0 문서 내에 있는 문서에 대한 참조는 참조되는 문서의 상태를 제공하지는 않는다.

1.3. 용어 정의

SAML 2.0 에 대해, 다음과 같은 용어 정의가 적용된다.

1.3.1. 들여온 정의들(Imported definitions)

1.3.1.1. SAML 2.0은 ITU-T Rec. X.667에서 정의된 다음 용어들을 사용한다:

a) UUID

1.3.1.2. SAML 2.0은 ITU-T Rec. X.680에서 정의된 다음 용어들을 사용한다:

a) 객체 식별자(Object identifier);

b) 오픈 타입 표기법(Open type notation).

1.3.1.3. SAML 2.0은 ITU-T Rec. X.811에서 정의된 다음 용어들을 사용한다:

사용자(Principle).

1.3.1.4. SAML 2.0은 ITU-T Rec. X.812에서 정의된 다음 용어들을 사용한다:

a) 접근 제어 정보(Access control information);

b) 사용자(User).

1.3.1.5. SAML 2.0은 W3C 웹 서비스 어휘에서 정의된 다음 용어들을 사용한다:

a) 초기 SOAP 송신자(Initial SOAP sender);

b) 네임스페이스(Namespace);

c) 최종 SOAP 수신자(Ultimate SOAP receiver);

d) XML 스키마(XML schema).

1.3.1.6. SAML 2.0은 IETF RFC 2828에서 정의된 다음 용어들을 사용한다:

a) 접근(Access);

b) 접근 제어(Access control);

c) 프락시(Proxy);

d) 프락시 서버(Proxy server);

f) 풀(Pull);

e) 푸시(Push);

g) 보안 아키텍처(Security architecture);

h) 보안 정책(Security policy);

i) 보안 서비스(Security service).

1.3.1.7. SAML 2.0은 IETF RFC 2396에서 정의된 다음 용어들을 사용한다:

a) Uniform resource identifier (URI);

b) URI 참조(URI reference).

1.3.2. 추가적인 용어 정의(Additional definitions)

1.3.2.1. 접근 권한(Access rights): 주체가 자원에 대하여 가질수 있는 인가된 상호작용의 타입을 설명. 예로는 읽기, 쓰기, 실행, 추가, 변경 그리고 삭제를 들 수 있다.

1.3.2.2. 계정(Account): 사용자와 비즈니스 서비스 제공자 사이에 정상적인 거래와 서비스를 제공하기 위한 형식적인 비즈니스 협약.

1.3.2.3. 계정 연결(Account linkage): 서로 다른 두 제공자에서 동일한 사용자를 나타내는 계정을 연관시키는 방법. 이를 통해 두 제공자들은 그 사용자에 대한 정보를 통신할 수 있다. 계정 연결은 속성 공유나 또는 Identity 연계(federation)을 통해 설정될 수 있다.

1.3.2.3. 능동적인 역할(Active role): 예를 들어 자원에 접근하는 등, 어떤 연산을 수행할 때, 시스템 엔티티가 가지는 역할.

1.3.2.4. 관리 도메인(Administrative domain): 하나 또는 그 이상의 관리 정책, 인터넷 도메인 이름 등록들, 공공 법률 엔티티들(예를 들어, 개인, 기업 또는 다른 조직), 호스트, 네트워크 디바이스 그리고 상호 연결되는 네트워크의 집합, 그리고 그들 위에서 동작하는 네트워크 서비스와 응용들의 어떠한 조합으로 정의되는 환경 또는 문맥. 관리 도메인은 하나 또는 그 이상의 보안 도메인을 포함하거나 또는 정의할 수 있다. 하나의 관리 도메인은 단일한 사이트 또는 다중 사이트를 포함할 수 있다. 관리 도메인을 정의하는 특징들은 시간이 지남에 따라 진화할 수 있다. 관리 도메인들은 관리 도메인 경계를 넘어 서비스를 제공하거나 또는 소비하는 것에 대하여 협약을 만들 수 있다.

1.3.2.5. 관리자(Administrator): 시스템을 설치하거나 또는 관리하는 사람 또는 시스템을 이용하여 시스템 엔티티, 사용자와/또는 내용을 관리하는 사람. 관리자는 일반적으로 특정 관리 도메인에 가입하게 되고 하나 이상의 관리 도메인에 가입할 수도 있다.

1.3.2.6. 가맹, 가맹 그룹(Affiliation, affiliation Group): 사용자(principal)에 대한 식별자들의 (연계

관점에서) 단일한 네임스페이스를 공유하는 시스템 엔티티 집합.

1.3.2.7. 익명성(Anonymity): 익명 상태. 이것은 이름이나 신원이 알려지거나 노출되지 않도록 하는 조건을 나타냄.

1.3.2.8. 보장하는 기관(Asserting party): 공식적으로, 하나 또는 그 이상의 SAML 기관을 호스트팅하는 관리 도메인. 비공식적으로, SAML 기관의 한 인스턴스.

1.3.2.9. 주장(Assertion): 주체에 대하여 수행되는 인증 행위, 주체에 대한 속성 정보 또는 명기된 자원에 대하여 주체가 행할 수 있는 인가 데이터 등에 대하여 SAML 기관이 생성한 데이터 조각.

1.3.2.10. 속성(Attribute): 객체의 독특한 특성. 실세계 객체에 대하여, 속성들은 종종 크기, 모양, 무게 및 색깔 등과 같은 물리적인 특징들로 명기된다. 사이버스페이스에서 객체는 크기, 인코딩 타입, 네트워크 주소 등등을 설명하는 속성들을 가질 수 있다. 속성들은 종종 “속성 이름”과 “속성 값(들)로 표현된다. 예를 들어, “foo”는 값 ‘bar’를 가지며, “count”는 값 1을, “gizmo”는 ‘frob”과 ‘2”를 값들로 가진다.

1.3.2.11. 속성 주장(Attribute assertion): 주체의 속성들에 대한 정보를 운반하는 주장.

1.3.2.12. 속성기관(Attribute authority): 속성 주장들을 생성하는 시스템 엔티티.

1.3.2.13. 인증(Authentication): 인증은 어떤 사람 또는 어떤 사물이 어느 정도의 신뢰 내에서 그것이 자신이 그렇다고 선언하는 것이 정말로 맞는지 아닌지를 결정하는 과정이다.

1.3.2.14. 인증 주장(Authentication assertion): 주체에 대하여 발생된 성공적인 인증 행위에 대한 정보를 운반하는 주장.

1.3.2.15. 인증 기관(Authentication authority): 인증 주장들을 생성하는 시스템 엔티티.

1.3.2.16. 인가(Authorization): 어떤 주체가 특정 자원에 대하여 명기된 타입의 접근을 수행하는 것이 허가되었는지를, 적용가능한 접근제어 정보를 평가함으로써, 결정하는 과정. 일반적으로, 인가는 인증 문맥 내에 있다. 일단 주체가 인증이 되면, 그것은 다른 타입들의 접근을 수행하는 것에 대하여 인가될 수 있다.

1.3.2.17. 인가 결정(Authorization decision): 인가 행위의 결과. 그 결과는 부정적인 될 수 있다. 즉, 그것은 주체가 자원에 대한 어떠한 접근 권한도 없음을 가리킨다.

1.3.2.18. 인가 결정 주장(Authorization decision assertion): 인가 결정에 대한 정보를 운반하는 주장.

1.3.2.19. 후 채널(Back channel): 후 채널은, 예를 들어 사용자 에이전트인 HTTP 클라이언트와 같은 또 다른 시스템 엔티티를 통하여 메시지를 리다이렉트(redirect) 하지 않고 두 시스템 엔티티들 사이에 직접적인 통신을 가리킨다.

1.3.2.20. 바인딩, 프로토콜 바인딩(Binding, protocol binding): 일반적으로, 어떤 프로토콜 메시지와 메시지 교환 패턴을 구체적인 방식으로 또 다른 프로토콜로 매핑시키는 것에 대한 명세임. 예를 들어, SAML <AuthnRequest> 메시지를 HTTP에 매핑하는 것은 바인딩의 한 예가 된다. 동일한 SAML 메시지를 SOAP으로 매핑하는 것은 또 다른 바인딩이 된다. SAML 문맥에서는, 각각의 바인딩에 “SAML xxx binding”이라는 패턴의 이름이 주어진다.

1.3.2.21. 크리덴셜(Credentials): 주장되는 사용자(principal) 신원을 확인하기 위해 전송되는 데이터.

1.3.2.22. 최종 사용자(End user): 응용 목적으로 자원을 사용하는 자연인(natural person).

1.3.2.23. **엔티티(Entity)**: “시스템 엔티티”를 참고한다.

1.3.2.24. **연계하다(Federate)**: 둘 또는 그 이상의 엔티티들을 함께 연결하거나 또는 바인딩하기.

1.3.2.25. **연계(Federation)**: 이 용어는 두가지 의미로 사용된다.:

1. 두 엔티티 사이에 관계를 설정하는 행위.
2. 어떠한 개수의 서비스 제공자들과 아이덴터티 제공자들로 구성된 하나의 연합(association).

1.3.2.26. **연계된 아이덴터티(Federated identity)**: 제공자들 사이에 그 사용자를 참조하기 위해 사용되는 식별자 집합과 속성들에 대하여 협정(agreement)가 있을 때, 사용자(principal)의 아이덴터티는 연계가 되었다고 말해진다.

1.3.2.27. **전 채널(Front channel)**: 전 채널은 두 개의 HTTP로 통신하는 서버들이 “HTTP redirect” 메시지를 채용하고 이를 통해, 예를 들어 웹 브라우저 또는 다른 어떠한 HTTP 클라이언트인 사용자 에이전트를 경유하여 상호간에 메시지를 전달하는 경우에 효과가 발생되는 통신 채널을 가리킨다.

1.3.2.28. **식별자(Identifier)**: 시스템 엔티티들 유일하게 가리키도록 시스템 엔티티에 매핑된 데이터 객체. 예를 들어 문자열이 될 수 있음. 시스템 엔티티는 그것을 가리키는 다중 식별자를 가질 수 있다. 하나의 식별자는 본질적으로 엔티티의 “구별되는 속성”이다.

1.3.2.29. **아이덴터티, 신원(Identity)**: 엔티티의 본질. 어떤 사물의 아이덴터티는 어떤 사물의 특징들로 종종 설명된다. 이 특성들 중에 식별자들이 포함될 수 있다.

1.3.2.30. **아이덴터티 탈연계(Identity defederation)**: 제공자들이 일정 집합의 식별자와/또는 속성들을 통해 사용자(principal)을 참조하는 것을 그만두기로 동의할 때, 발생하는 동작.

1.3.2.31. **아이덴터티 연계(Identity federation)**: 사용자(principal)을 위해 연계된 아이덴터티를 생성하는 동작.

1.3.2.32. **아이덴터티 제공자(Identity provider)**: 사용자(principal)들을 위해 아이덴터티 정보를 생성하고, 유지하며, 관리하고 그리고 웹 브라우저 프로파일과 같이 하나의 연계 내에서 다른 서비스 제공자에게 사용자(principal) 인증을 제공하는 일종의 서비스 제공자.

1.3.2.33. **아이덴터티 제공자 라이트(Identity provider lite)**: 단지 SAML에서 요구되는 부분만을 사용하여, 사용자(principal)들을 위해 아이덴터티 정보를 생성하고, 유지하며, 관리하고 그리고 웹 브라우저 프로파일과 같이 하나의 연계 내에서 다른 서비스 제공자에게 사용자(principal) 인증을 제공하는 일종의 서비스 제공자.

1.3.2.34. **로그인, 로그온, 사인-온(Login, logon, sign-on)**: 일종의 처리. 이 처리를 통해 사용자가 인증기관에게 크리덴셜을 제출하고 간단한 세션을 설정하고 그리고 선택적으로 리치(rich) 세션을 설정한다.

1.3.2.35. **로그아웃, 로그오프, 사인-오프(Logout, logoff, sign-off)**: 일종의 처리. 이 처리를 통해 사용자는 단순 세션 또는 리치(rich) 세션을 종료하기를 원한다는 것을 알린다.

1.3.2.36. **마크업 언어(Markup language)**: 특수한 목적으로 XML 문서의 구조에 적용되는 일단의 XML 요소들과 XML 속성들. 마크업 언어는 일반적으로 일단의 XML 스키마들과 동반되는 문서로 정의된다.

1.3.2.37. **이름 제한자(Name qualifier)**: 다른 사용자들(principals)을 나타내기 위해, (연계

관점에서) 하나 이상의 네임스페이스에서 사용될 수 있는 하나의 식별자가 모호해지지 않도록 해주는 문자열.

1.3.2.38. 기관, 당사자(Party): 비공식적으로, 주장을 수신하거나 또는 자원을 접근하는 것과 같은 어떤 처리나 통신에 참여하는 하나 또는 그 이상의 사용자들(principals).

1.3.2.39. 영속적인 의사익명(Persistent pseudonym): 다중 세션에 걸쳐있는 확장된 기간 동안에 주어진 의지하는 기관이 사용자를 식별할 수 있도록, 어떤 아이덴터티 제공자에 의해 할당된 프라이버시-보호형 이름 식별자. 아이덴터티 연계를 나타내는데 사용될 수 있다.

1.3.2.40. 정책 결정점(Policy decision point (PDP)): 자신을 위해 인가 결정을 내리거나 또는 이와 같은 결정을 요구하는 다른 시스템 엔티티를 위해 인가 결정을 내리는 시스템 엔티티. 예를 들어, SAML PDP는 인가 결정 요청들을 받아들여, 응답으로 인가 결정 주장들을 생성한다. PDP는 인가 결정 기관이다.

1.3.2.41. 정책 집행점(Policy enforcement point (PEP)): 인가 결정을 요청하고 뒤이어 집행하는 시스템 엔티티. 예를 들어, SAML PEP는 일가 결정 요청들을 PDP에게 전달하고, 응답으로 수신되는 인가 결정 주장들을 처리한다.

1.3.2.42. 사용자 아이덴터티(Principal identity): 일반적으로 식별자인 어떤 사용자 아이덴터티의 표현.

1.3.2.43. 프로파일(Profile): 여러 목적 중에 하나를 위한 일단의 규칙들. 각각의 집합은 “SAML xxx 프로파일” 또는 “xxx SAML 프로파일” 패턴으로 이름이 주어진다.

1. 어떤 프로토콜 또는 다른 사용 문맥에 주장을 내장시키거나 또는 그것들로부터 추출하는 방법에 대한 규칙들.
2. 특수한 사용 문맥에서 SAML 프로토콜 메시지를 사용하는 것에 대한 규칙들.
3. SAML 로 표현된 속성들을 또 다른 속성 표현 시스템으로 매핑시키는 것에 대한 규칙들. 이와 같은 규칙의 집합은 “속성 프로파일”로 알려진다.

1.3.2.44. 프로토콜 바인딩(Protocol binding): “바인딩”을 참고한다.

1.3.2.45. 제공자(Provider): 아이덴터티 제공자들과 서비스 제공자들 둘 다를 가리키는 포괄적인 표현.

1.3.2.46. 의지하는 기관(측)(Relying party): 다른 시스템 엔티티가 제공한 정보를 기반으로 행동을 취할 것을 결정하는 시스템 엔티티. 예를 들어, SAML 의지하는 기관은 주체에 대하여 보장하는 기관(SAML 기관)이 제공한 주장들을 의지한다.

1.3.2.47. 요청자(Requester): 또 다른 시스템 엔티티(SAML 기관, 응답자)에게 서비스를 요청하기 위해 SAML 프로토콜을 활용하는 시스템 엔티티. 많은 시스템 엔티티들이 클라이언트와 서버 둘 모두로서 동시에 또는 순차적으로 동작하기 때문에, 이 표시법에서 “클라이언트” 라는 용어는 사용되지 안 된다. SAML SOAP 바인딩이 사용중인 경우에는, SAML 요청자는 초기 SOAP 송신자와 구조적으로 분리된다.

1.3.2.48. 자원(Resource): (예를 들어, 파일 형태나 메모리 형태, 등등으로) 하나의 정보 시스템에 포함되는 데이터, 또한:

1. 시스템이 제공하는 서비스.

2. 시스템 장비의 한 항목(다른 말로, 하드웨어, 펌웨어, 소프트웨어 또는 문서등과 같은 시스템 컴포넌트)

1.3.2.49. 응답자(Responder): 또 다른 시스템 엔티티(요청자)로부터 전달받은 서비스 요청에 대하여 응답하기 위해 SAML 프로토콜을 활용하는 시스템 엔티티(SAML 기관). 많은 시스템 엔티티들이 클라이언트와 서버 둘 모두로서 동시에 또는 순차적으로 동작하기 때문에, 이 표시법에서 “서버”라는 용어는 사용되지 안 된다. SAML SOAP 바인딩이 사용중인 경우에는, SAML 응답자는 최종 SOAP 수신자와 구조적으로 분리된다.

1.3.2.50. 역할, 롤(Role): 사전들은 역할을 “수행자에 의해 동작되는 특성” 또는 “함수 또는 위치)로 정의한다. 시스템 엔티티들은 예를 들어 능동적인 역할들과 수동적인 역할들과 같은 다양한 타입들의 역할들을 순차적으로/또는 동시적으로 수행한다. 관리자의 개념은 종종 역할의 한 예이다.

1.3.2.51. SAML 아티팩트(SAML artifact): 일반적으로 더 크고, 가변-크기의 SAML 프로토콜 메시지를 가리키는 작고, 고정-크기를 가지는 구조화된 데이터 객체. SAML 아티팩트들은 “3xx Redirection” 상태 코드들을 가지는 HTTP 응답 메시지들과 뒤따르는 HTTP GET 메시지들과 같이 URL에 내장되고 HTTP 메시지들을 통해 운반되도록 설계된다. 이런 방식으로, 서비스 제공자는 간접적으로, 사용자 에이전트를 경유하여, 다른 제공자에게 SAML 아티팩트를 전달할 수 있다. 다른 제공자는 artifact를 제공하는 제공자와의 직접적은 상호작용을 통해 SAML 아티팩트를 디레퍼런스(dereference)하여 SAML 프로토콜 메시지를 얻을 수 있다.

1.3.2.52. SAML 기관(SAML authority): SAML 도메인 모델에서 주장들을 발급하는 추상적인 시스템 엔티티. 속성 기관, 인증 기관, 정책 결정점(PDP)를 또한 참고한다.

1.3.2.53. 보안(Security): 정보의 기밀성을 보장하고, 그것을 처리하는데 사용되는 시스템과 네트워크를 보호하고, 그들에 대한 접근을 제어하는 일단의 보호방법들. 보안은 일반적으로 비밀(secrecy), 기밀성, 무결성, 이용가능성 등의 개념을 포괄한다. 이것은 어떤 시스템이 잠재적으로 상호연관된 공격들을 방어하는 것을 보장하기 위한 것이다.

1.3.2.54. 보안 주장(Security assertion): 보안 아키텍처의 문맥에서 철저히 검사된 주장.

1.3.2.55. 보안 문맥(Security context): 개별적인 SAML 프로토콜 메시지에 대하여, 메시지의 보안 문맥은 만약 있다면 메시지의 보안 헤더 블록들과 수신자에게 메시지를 배달할 때, 사용될 수 있는 다른 보안 메커니즘들의 의미적인 합(semantic union)이다. HTTP, TLS와 IPSEC등과 같은 하부 네트워크 스택 레이어들에서 채택되는 보안 메커니즘들이 후자의 예가 된다.

1.3.2.56. 보안 도메인(Security domain): 일단의 자원들과 그들 자원들을 접근하는 것이 인가된 시스템 엔티티들을 포함하여, 보안 모델과 보안 아키텍처에서 정의된 환경 또는 문맥. 하나 또는 그 이상의 보안 도메인들이 단일 관리 도메인(administrative domain)에 존재할 수 있다. 어떠한 보안 도메인을 정의하는 특징들은 시간이 지남에 따라 일반적으로 진화한다.

1.3.2.57. 보안 정책 표현(Security policy expression): 사용자(principal) 아이덴터티들과 또는 그것의 속성들을 허용가능한 동작들(actions)로 매핑하는 것. 보안 정책 표현은 종종 본질적으로 접근 제어 리스트가 된다.

1.3.2.58. 서비스 제공자(Service provider): 어떤 시스템 엔티티에게 주어진 역할. 이 역할을 통해 그 시스템 엔티티는 사용자들(principals) 또는 다른 시스템 엔티티들에게 서비스들을 제공한다.

1.3.2.59. 서비스 제공자 라이트(Service provider lite): 어떤 시스템 엔티티에게 주어진 역할. 이 역할을 통해 그 시스템 엔티티는 단지 필요한 SAML 프로토콜 부분만을 사용하여, 사용자들(principals) 또는 다른 시스템 엔티티들에게 서비스들을 제공한다.

1.3.2.60. 세션(Session): 상호작용 기간 동안 상호작용에 대한 일부 상태를 유지하는 것을 특징으로 하는, 종종 사용자를 포함하는(Principal), 시스템 엔티티들의 지속적인 상호작용.

1.3.2.61. 세션 기관(Session authority): 세션들과 관련된 상태를 어떤 시스템 엔티티가 유지할 때, 그 기관에게 주어진 역할.

1.3.2.62. 세션 참여자(Session participant): 어떤 기관이 적어도 하나의 세션 기관과 어떤 세션에 참여할 때, 그 기관에게 주어진 역할.

1.3.2.63. 사인-오프(Sign-off): "로그아웃"을 참고한다.

1.3.2.64. 사인-온(Sign-on): "로그인"을 참고한다.

1.3.2.65. 사이트(Site): 지리적인 또는 DNS 이름 관점에서 하나의 관리 도메인을 나타내는 비공식적인 용어. 이것은 어떤 관리 도메인의 특정 지리적인 또는 위상적인(topological) 부분을 나타낼 수도 있고, 또는 하나의 ASP 사이트에서 그렇듯이, 다중 관리 도메인들을 포괄할 수도 있다.

1.3.2.66. 주체(Subject): 어떤 보안 도메인 문맥에서 하나의 사용자(principal). SAML 주장들은 주체에 대한 선언들을 생성한다.

1.3.2.67. 시스템 엔티티, 엔티티(System entity, entity): 컴퓨터/네트워크 시스템의 능동적인 어떤 요소. 예를 들어, 자동화된 처리 또는 처리 집합. 하부 시스템, 분리된 기능 집합을 통합하는 사람 또는 사람들 그룹.

1.3.2.68. 타임-아웃(Time-out): 만약 어떤 사건이 발생하지 않았다면, 그 시각 이후, 어떤 조건이 "참"이 되는 기간. 예를 들어, 세션의 상태가 특정 기간 동안 비활성화되어 있었기 때문에 종료되는 세션은 "타임 아웃" 되었다고 말해진다.

1.3.2.69. 일시적인 의사익명(Transient pseudonym): 다중 세션에 걸쳐있을 필요가 없는 상대적으로 짧은 기간 동안에 주어진 의지하는 기관이 사용자를 식별할 수 있도록, 어떤 아이덴터티 제공자에 의해 할당된 프라이버시-보호형 이름 식별자.

1.3.2.70. XML 요소(XML attribute): XML 요소의 시작-태그(start-tag)에 포함되어 있고, 이름과 값을 가지는 XML 데이터 구조.

1.3.2.71. XML 요소(XML element): XML 문서 내에서 다른 이와 같은 구조들 사이에서 구조적으로 배열되며, 시작-태그(start-tag)와 종료-태그(end-tag) 또는 빈 태그(empty tag)로 가리켜지는 XML 데이터 구조.

1.4. 약어(Abbreviations)

AA	Attribute Authority
ASN.1	Abstract Syntax Notation One
ASP	Application Service Provider
CA	Certification Authority

CMP Certificate Management Protocol

CRL Certificate Revocation List

DDDS Dynamic Delegation Discovery System

DCE Distributed Computing Environment

DNS Domain Name System

ECP Enhanced Client/Proxy

HTTP HyperText Transfer Protocol

HTTPS Secure HyperText Transport Protocol

IdP Identity Provider

IdP Lite Identity Provider Lite

IP Internet Protocol

IPSEC Internet Protocol SECurity

MD5 Message Digest algorithm 5

MIME Multipurpose Internet Mail Extensions

NAPTR Naming Authority PoinTeR

OID Object IDentifier

PAC Privilege Attribute Certificates

PAOS Reverse SOAP

PDP Policy Decision Point

PEP Policy Enforcement Point

PGP Pretty Good Privacy

PKI Public-Key Infrastructure

POP Proof Of Possession

RA Registration Authority

RSA Rivest Shamir Adleman public key algorithm

SHA-1 Secure Hash Algorithm 1

SP Service Provider

SPKI Simple Public Key Infrastructure

SP Lite Service Provider Lite

SSO Single Sign On

TLS Transport Layer Security protocol

URI Uniform Resource Identifier

UTC Coordinated Universal Time

UUID Universal Unique IDentifier

XACML eXtensible Access Control Markup Language

XML eXtensible Markup Language

1.5. 관례(Conventions)

SAML 2.0에서 사용되는 키워드인 "해야만 한다(must)", "하지 않아야만 한다(must not)", "요구된다(required)", "일 것이다(shall)", "이지 않을 것이다(shall not)", "해야 한다(should)", "하지 않아야 한다(should not)", "권고된다(recommended)", "일(할) 수 있다(may)", "선택적인(optional)"은 IETF RFC 2119에서 설명된 것과 같이 해석되어야 한다.

SAML 2.0은 W3C XML 스키마 Part 1, W3C 스키마 Part 2와 그들 표준들의 규범적 텍스트(normative text)를 사용하여 XML 인코딩된 SAML 주장과 프로토콜 메시지들의 문법과 의미를 설명한다. SAML 2.0의 SAML 스키마 문서들과 스키마 리스트 사이에 불일치가 발생할 경우에는, 스키마 문서가 높은 우선순위를 가진다. 어떤 경우에는, SAML 2.0이 스키마 문서에 의해 가리키는 것 이상의 제약을 가하는 경우가 있다는 것에 주의해야 한다.

1.6. 공통 데이터 타입(Common data types)

다음 하부 절들은 SAML 스키마들에서 나타나는 공통된 데이터 타입들을 어떻게 사용하고 해석하는지를 정의한다.

1.6.1. 문자열 값(String Values)

모든 SAML 문자열 값들은 **xs:string** 타입을 가지며, 이 타입은 W3C XML 스키마 데이터타입들 표준에 내장(built in) 되어 있다. SAML 2.0 에서 별다른 언급이 없으면, SAML 메시지들에 존재하는 모든 문자열들은 적어도 하나 이상의 공백이 아닌 문자(non-whitespace)로 구성되어야만 한다.

이 SAML 2.0 또는 특정 프로파일들에서 별다른 언급이 없으면, XML 스키마 **xs:string** 타입을 가지거나 또는 이 문자열 타입으로부터 유도된 타입을 가지는 SAML 문서 내의 모든 요소들은 정확한 이진 비교(exact binary comparison)를 사용하여 비교되어야만 한다. 특히, SAML 구현과 배치(deployment)들은 대소문자를 구분하지 않는 문자열 비교, 공백의 정규화 또는 절단(trimming) 또는 숫자나 화폐와 같이 로케일에 따라 고유한(locale-specific) 변환 등에 의존하지 않아야만 한다. 이 요구는 W3C 문자열의 요구사항을 따르게 하기 위해 의도된 것이다.

만약 어떤 구현이 다른 문자 인코딩(encodings) 방식들을 사용하여 표현된 값들을 비교한다면, 그 구현은 두 값을 유니코드 문자 인코딩인 정규화 폼 C(Normalization Form C)로 변환하고 그것들에 대하여 정확한 이진 비교를 수행한 것과 같은 결과를 반환하는 비교 방법을 사용해야만 한다. 이 요구는 W3C 문자 모델과 특히, 유니코드-정규화 텍스트(Unicode-normalized Text)들에 대한 규칙을 따르게 하기 위해 의도된 것이다.

SAML 문서 형태로 받은 데이터와 외부 소스로부터 받은 데이터를 비교하는 응용(application)은 XML 에 대해 규정된 정규화 규칙을 고려해야만 한다. 요소들 내에 포함된 텍스트(text)는 라인의 끝이 라인피드 문자들(ASCII code $10_{Decimal}$)을 사용하여 나타내도록 정규화된다. 문자열들 (또는 문자열로부터 유도된 타입들)로 정의된 XML 속성 값들은 W3C XML 1.0, 3.3.3 절에서 설명된 것처럼 정규화된다. 모든 공백 문자들은 스페이스(blanks) (ASCII code $32_{Decimal}$)로 대체된다.

SAML 2.0 은 XML 속성 값들 또는 요소 내용에 대하여 대조(collation) 또는 정렬 순서를 정의하지 않는다. SAML 구현들은 값들에 대하여 특정한 정렬 순서들에 의존하지 않아야만 한다. 왜냐하면 처리에 참여한 호스트(host)들에서 설정된 로케일(locale)에 따라, 그 정렬 순서들이 달라지기 때문이다.

1.6.2. URI 값(URI Values)

모든 SAML URI 참조 값들은 **xs:anyURI** 타입을 가지며, 이 타입은 W3C XML 스키마 데이터타입들에 내장(built in) 되어 있다.

SAML 2.0 에서 다르게 지시되지 않는다면, SAML 에서 정의된 속성들 또는 요소들 내에서 사용되는 모든 URI 참조 값들은 적어도 하나 이상의 공백이 아닌 문자로 구성되어야만 하며, 절대경로를 표현하도록 요구된다.

SAML 2.0 은 상태코드, 포맷 타입, 속성과 시스템 엔티티 이름들 등과 같은 식별자들로써 URI 참조를 광범위하게 사용한다. 따라서, 똑 같은 URI 가 다른 시각에 다른 정보를 나타내는데 절대로 사용되지 않도록, URI 값들이 유일하고 동시에 일관되도록(consistent) 하는 것이 필수적이다.

1.6.3. 시간 값(Time Value)

모든 SAML 의 시각 값들은 **xs:dateTime** 타입을 가지며, 이 타입은 W3C XML 스키마 데이터타입들에 내장(built in) 되어 있다. 모든 SAML 시각 값들은 시간대(time zone) 컴포넌트가 없는 UTC 형식(form)으로 표현되어야만 한다.

SAML 시스템 엔티티들은 1000 분의 1 초보다 더 정교한 시각에 의존하지 않아야 한다. 구현들은 윤초(leap seconds)를 명기하는 시각 값들을 생성하지 않아야만 한다.

1.6.4. ID 와 ID 참조 값(ID and ID Reference Values)

xs:ID 단순 타입은 주장들, 요청 및 응답에 대한 SAML 식별자들(identifiers)을 선언하는데 사용된다. SAML 2.0에서 **xs:ID** 타입으로 선언된 값들은 **xs:ID** 타입 자체의

정의에 의해 주어진 특성뿐만 아니라 다음과 같은 특성들을 만족시켜야만 한다:

- 식별자들은 할당하는 어떠한 기관(party)도 자신 또는 다른 기관(party)이 다른 데이터 객체에게 우연히 동일한 식별자를 할당할 수 있는 가능성이 거의 무시할 수 있을 정도라는 것을 보장해야만 한다.
- 어떤 데이터 객체가 자신이 특정한 식별자를 가지고 있다고 선언한 곳에, 그와 같은 선언은 정확히 하나만 있어야만 한다.

SAML 시스템 엔티티가 그것이 생성하는 식별자가 유일하다는 것을 보장하는 메커니즘은 시스템 구현에 의해 결정된다. 랜덤(random) 또는 의사랜덤(pseudorandom) 기술이 채택된 경우에, 임의적으로 선택된 두 개의 식별자가 서로 동일할 확률은 2^{-128} 보다 작거나 같아야만 하고, 2^{-160} 보다 작거나 같아야 한다. 이 요구는 128 비트와 160 비트 사이의 길이를 갖는 임의적으로 선택된 값을 인코딩함으로써 충족될 수 있다. 인코딩은 xs:ID 데이터타입을 정의하는 규칙을 준용해야만 한다. 의사랜덤 발생기는 서로 다른 시스템들 사이에 바람직한 유일성 특성을 보장하기 위해 유일한 값(material)으로 시드(seed)를 설정하여야만 한다.

2. 프로파일 개념

이 문서는 통신 프로토콜과 프레임워크에서 SAML 주장과 요청-응답 메시지를 사용하는 것을 정의하는 프로파일을 명기한다. 또한, 이 문서는 SAML 속성 값 문법과 명명 규정들(naming conventions)을 정의하는 프로파일을 명기한다.

2.1. 프로파일 개념

어느 한 타입의 SAML 프로파일은 프레임워크나 프로토콜에 SAML 주장들을 삽입하고 추출하는 방법을 설명하는 규칙들에 대하여 개략적으로 설명한다(outline). 그러한 프로파일은 메시지를 최초 송신자(originating party)에서 어떻게 SAML 주장을 다양한 타입의 파일 또는 통신 프로토콜의 프로토콜 데이터 유닛 등과 같은 다른 객체에 삽입하거나 또는 조합하는지를 설명하고, 어떻게 SAML 주장이 최초 송신자에서 수신자(receiving party)로 전달되고, 그 다음에 최종 목적지에서 처리되는지를 설명한다. 특정 클래스 <FOO> 객체들에 SAML 주장들을 삽입하거나 또는 그 객체들로부터 추출하는 것에 대한 특정 규칙들의 집합을 SAML 의 <*FOO*> 프로파일이라고 부른다.

예를 들어, SAML 의 SOAP 프로파일은 어떻게 SAML 주장들이 SOAP 메시지에 추가되는지, 어떻게 SAML 주장들이 SOAP 헤더들에게 영향을 미치는지 그리고 어떻게 SAML 과 관련된 에러 상태들이 SOAP 메시지에 반영되어야 하는지에 대하여 설명한다.

또 다른 타입의 SAML 프로파일은 특정 환경이나 사용 문맥에서 일반적인 SAML 프로토콜이나 또는 주장 기능의 사용에 대한 일단의 제약들을 정의한다. 이러한 특징을 갖는 프로파일들은 선택권(optionality)를 제약할 수 있고, 예를 들어 속성들, 조건들 또는 바인딩과 같은 특정 SAML 기능의 사용을 요구할 수 있고, 그리고 다른 측면에서 프로파일 행위자(profile actors)가 따라야 하는 처리 규칙들을 정의한다.

후자의 특정 예는 SAML 속성들을 가리키는 프로파일들을 들 수 있다. SAML <Attribute> 요소는 속성 명명, 값 문법 그리고 XML 속성들의 사용을 통해 대역내(in-band) 메타데이터를 포함하는 것에 대하여 많은 유연성(flexibility)를 제공한다. 타입은 SAML 2.0 주장과 프로토콜 표준에서 정의된 일반적인 규칙보다 더 많이 구체적으로 어떻게 이들 요소들을 사용할지를 정의하는 프로파일들을 준수함으로써 이러한 유연성을 제약할 수 있게 되고, 이를 통해 상호운용성이 달성된다.

속성 프로파일들은 특정 타입의 속성 정보를 다룰 때나, 조금 더 엄격한 사용(strictness)를 요구하는 외부 시스템들이나 다른 공개 표준들과 상호동작을 할 때, SAML 속성 표현을 제약하는데 필요한 정의들을 제공한다.

이 표준은 독립적으로 구현된 시스템들이 상호운용될 수 있다는 것을 보장하는데 필요한 만큼, 선택된 다양한 종류의 프로파일들을 충분히 그리고 자세히 명기하는 것을 목적으로 한다.

3. 추가적인 프로파일들의 명기

이 표준에서는 일단의 선택된 프로파일들을 정의한다. 그러나, 향후 다른 프로파일들이 개발되는 것 역시 가능하다. 다음 절들은 추가적인 프로파일들을 명기하기를 원하는 제 3 자를 위한 가이드라인을 제공한다.

3.1. 프로파일 명기를 위한 가이드라인

다음은 각각의 프로파일에서 반드시 기술되어야만 하는 이슈들의 체크 리스트이다.

1. 프로파일을 유일하게 식별하는 URI, 저자에 대한 우편 또는 전자적인 연락 정보를 명기한다. 그리고 새로운 프로파일을 갱신하거나 또는 폐기시키기 이전에 정의된 프로파일에 대한 참조를 제공한다.
2. 프로파일에 연관된 당사자들(parties) 사이의 상호작용들을 설명한다. 각각의 당사자에 의해 사용되는 응용(applications)과 각각의 상호작용에 관련된 프로토콜들에 대한 어떠한 제약들도 명확히 선언되어야 한다.
3. 얼마나 많은 당사자들이 관련되어 있고 중개자들(intermediaries)이 관련되어 있는지를 포함하여, 각각의 상호작용에 관련된 당사자들을 식별한다(identify).

4. 인증이 필요한지 여부와 수용할 수 있는 인증 타입들을 포함하여, 각각의 상호작용에 연관된 당사자들의 인증 방법을 명기한다.
5. 메시지 무결성을 보장하는데 사용된 메커니즘들을 포함하여, 메시지 무결성에 대한 지원 수준(level)을 식별한다.
6. 제 3자가 SAML 메시지와 주장의 내용을 볼 수 있는지 여부와, 프로파일이 기밀성(confidentiality)를 요구하는지 여부, 그리고 기밀성을 달성하기 위해 권고되는 메커니즘들을 포함하여, 기밀성에 대한 지원 수준을 식별한다.
7. 특히 SAML 주장 또는 메시지를 받고 처리하는 과정에서 발생하는 에러 상태들을 포함하여, 각각의 참여자에서 발생하는 에러 상태들 포함하는 에러 상태를 식별한다.
8. 위협들에 대한 분석과 대처방안(countermeasures)를 포함하여, 보안 고려사항들을 식별한다.
9. 프로파일에서 정의되고/또는 활용되는 확인 방법 식별자(confirmation method identifier)를 식별한다.
10. 프로파일에서 정의되고/또는 활용되는, 관련된 SAML 메타데이터를 식별한다.

3.2. 속성 프로파일 명기를 위한 가이드라인

다음은 속성 프로파일들에서 특히 기술되어야만 하는 이슈들의 체크 리스트이다.

1. 프로파일을 유일하게 식별하는 URI, 저자에 대한 우편 또는 전자적인 연락 정보를 명기한다. 그리고 새로운 프로파일을 갱신하거나 또는 폐기시키기 이전에 정의된 프로파일에 대한 참조를 제공한다.
2. SAML <Attribute> 요소들의 NameFormat과 Name 속성이 가질 수 있는 값에 대한 문법과 제약사항들.
3. 프로파일에서 정의되어 SAML <Attribute> 요소들에서 사용될 수 있는 어떠한 추가적인 네임스페이스-제한된(namespace-qualified) XML 속성들.
4. 속성들, 질의들 및 기타 등등을 처리할 때 사용될 수 있도록, 프로파일에서 정의된 SAML <Attribute> 요소들의 동등성(equality)을 결정할 수 있는 규칙들.
5. xsi:type XML 속성이 사용될 수 있거나 또는 사용되어야 하는지 여부를 포함하여, SAML <AttributeValue> 요소가 가질 수 있는 값에 대한 문법과 제약사항들

4. 확인 방법 식별자(Confirmation method identifiers)

SAML 주장과 프로토콜 표준은 Method로써 <SubjectConfirmation> 요소와 추가적으로 선택요소인 <SubjectConfirmationData>를 정의한다. <SubjectConfirmation> 요소는 의지하는 측(relying party)에서 요청이나 메시지가 특정 프로파일 문맥 내에서 주장의 주체와 연관된 시스템 엔티티로부터 온 것인지를 확인하기 위해 사용되어야 한다.

Method 속성은 의지하는 측이 이러한 결정을 내리는데 사용해야 하는 방법을 가리킨다. 이것은 이전에 수행했던 인증과 어떠한 연관성을 가질 수도 있고 또는 가지지 않을 수도 있다. 인증 문맥과는 달리, 주체 확인 방법은 <SubjectConformationData> 요소에 인증서 또는 키와 같은 추가적인 정보를 수반할 것이다. <SubjectConformationData> 요소는 의지하는 측이 필요한 검증(verification)을 수행할 수 있도록 해 준다. 공통의 속성 집합이 또한 정의되며, 이들은 검증이 이루어질 수 있는 조건들을 제약하는데 사용될 수 있다.

프로파일들은 <ConfirmationMethod>를 위해 여러 가지 다른 값들을 정의하고 사용할 것으로 예상된다. 이들 각각의 값들은 각기 다른 SAML 사용 시나리오에 대응된다. 이 표준에서 정의된 프로파일들과 이 표준 밖의 다른 프로파일들에서 유용하게 사용할 수 이있는 다음 방법들을 정의한다.

4.1. 키 소유자(Holder of Key)

URI: urn:oasis:names:tc:SAML:2.0:cm:holder-of-key

하나 또는 그 이상의 <ds:KeyInfo> 요소들이 <SubjectConfirmationData> 요소 내에 존재해야만 한다. xsi:type 속성이 <SubjectConfirmationData> 요소에 존재할 수 있으며, 만약 존재한다면, 반드시 **saml:KeyInfoConfirmationDataType**으로 설정되어야만 한다. **saml:** 네임스페이스 접두사는 어떤 것이 와도 좋지만, 반드시 SAML 주장 네임스페이스를 참조해야만 한다.

W3C Signature에서 설명된 것처럼, 각각의 <ds:KeyInfo> 요소는 키 자체나 또는 응용이 키를 얻을 수 있도록 하는 정보를 포함한다. 명기된 키의 소유주가 보장하는 측에 의해 생성된 주장의 주체와 동일한 것으로 간주된다.

W3C Signature에 부합하도록, 각각의 <ds:KeyInfo> 요소는 반드시 단 하나의 암호 키를 식별해야만 한다는 것에 주의한다. 다른 확인 키들이 다른 의지하는 측들(relying parties)을 위해 필요한 때와 같은 경우에는, 다중 키가 분리된 <ds:KeyInfo> 요소들에서 식별될 수 있다.
예: “By-Tor”로 명명된 키의 소유주 또는 “Snow Dog”으로 명명된 키의 소유주는 그 자신이 주장의 주체라는 사실을 확인시킬 수 있다.

```
<SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:holder-of-key">
```

```
    <SubjectConfirmationData xsi:type="saml:KeyInfoConfirmationDataType">
        <ds:KeyInfo>
            <ds:KeyName>By-Tor</ds:KeyName>
        </ds:KeyInfo>
        <ds:KeyInfo>
            <ds:KeyName>Snow Dog</ds:KeyName>
        </ds:KeyInfo>
    </SubjectConfirmationData>
</SubjectConfirmation>
```

4.2. 송신자 단언들(Sender Vouches)

URI: urn:oasis:names:tc:SAML:2.0:cm:sender-vouches

주장의 사용 문맥에 대한 어떠한 다른 정보도 이용 가능하지가 않다는 것을 가리킨다. 의지하는 측(relying party)은 주장을 더 이상 처리해야 하는지 여부를 결정할 수 있는 다른 수단들을 활용해야 한다. 이 작업은 <SubjectConfirmationData>에 존재할 수 있는 속성들을 사용하여 확인에 대한 선택적인 제약 사항들에 영향을 받는다.

4.3. 운반자(Bearer)

URI: urn:oasis:names:tc:SAML:2.0:cm:bearer

주장의 주체는 주장의 운반자가 된다. 이 때 SAML 주장과 프로토콜 표준에서 정의된 것처럼, <SubjectConfirmationData>에 존재할 수 있는 속성들을 사용하여 확인에 대한 선택적인 제약 사항들에 영향을 받는다.

예: 주장이 "_1234567890"를 ID로 갖는 요청에 대한 응답으로, 2004년 3월 19일 GMT 오후 1:37 전에 "https://www.serviceprovider.com/saml/consumer"에 보내진 메시지를 통해 배달되었다면, 주장의 운반자는 그 자신이 주장의 주체라는 사실을 확인시킬 수 있다.

```
<SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
    <SubjectConfirmationData
            InResponseTo="_1234567890"
            Recipient="https://www.serviceprovider.com/saml/consumer"
            NotOnOrAfter="2004-03-19T13:27:00Z"
    </SubjectConfirmationData>
</SubjectConfirmation>
```

5. SAML SSO 프로파일

일단의 프로파일들이 브라우저들과 다른 클라이언트 장치들의 SSO(Single Sign-On)을 지원하기 위해 정의된다.

- SAML 주장과 프로토콜 표준에서 정의된 인증 요청 프로토콜에 대한 웹 브라우저 기반 프로파일은 웹 SSO를 지원하기 위해 정의된다. 이것은 최초의(original) SAML 요청 문서의 시나리오 1-1을 지원한다.
- 추가적인 웹 SSO 프로파일은 개선된 클라이언트(enhanced clients)를 지원하기 위해 정의된다.
- SAML 2.0 주장과 프로토콜 표준에서 정의된 단일 로그아웃과 이름 식별자 관리 프로토콜에 대한 프로파일은 전-채널(front-channel) (브라우저)와 후-채널(back-channer) 바인딩 둘 모두에서 정의된다.
- 추가적인 프로파일이 쿠키(cookies)를 사용하여 IdP(Identity Provider) 발견(discovery)을 위해 정의된다.

5.1. Web Brower SSO 프로파일

웹 브라우저(Web Browser) SSO 프로파일에서 지원되는 시나리오에서, 웹 사용자는 서비스 제공자(SP, Service Provider)의 자원을 접근하거나, 또는 SP 와 사용자가 원하는 자원이 이해하는 IdP 를 접근한다. 만약, 웹 사용자가 IdP 에게 인증되지 않았다면, 먼저 사용자는 IdP 에게 인증 받는다. 사용자가 이미 IdP 에게 인증을 받았다면, 이 과정은 생략될 수 있다. 이 과정은 IdP 가 인증 주장을 생성하도록 한다. 인증 주장을 생성할 때는 SP 로부터 일정 부분의 입력을 받을 수 있다. 이렇게 생성된 인증 주장은 SP 가 활용하여 웹 사용자를 위해 보안 문맥을 설정하는데 사용된다. 이러한 처리 과정 중에, IdP 와 SP 들 간에 사용자(principal)에 대한 이름 식별자가 생성될 수 있다. 이 작업은 상호작용 파라미터들과 당사자들의 동의에 영향을 받는다.

이 시나리오를 구현하기 위해, SAML 인증 요청 프로토콜에 대한 프로파일이 사용된다. 이 프로파일에서는 HTTP Redirect, HTTP POST 그리고 HTTP Artifact 바인딩이 결합되어 사용된다.

사용자는 표준적인 상용 브라우저를 이용하고 있으며, SAML 영역 밖의 어떤 수단들을 통해 IdP 게 인증할 수 있다는 것을 가정한다.

5.1.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:SSO:browser

Contact information: security-services-comment@lists.oasis-open.org

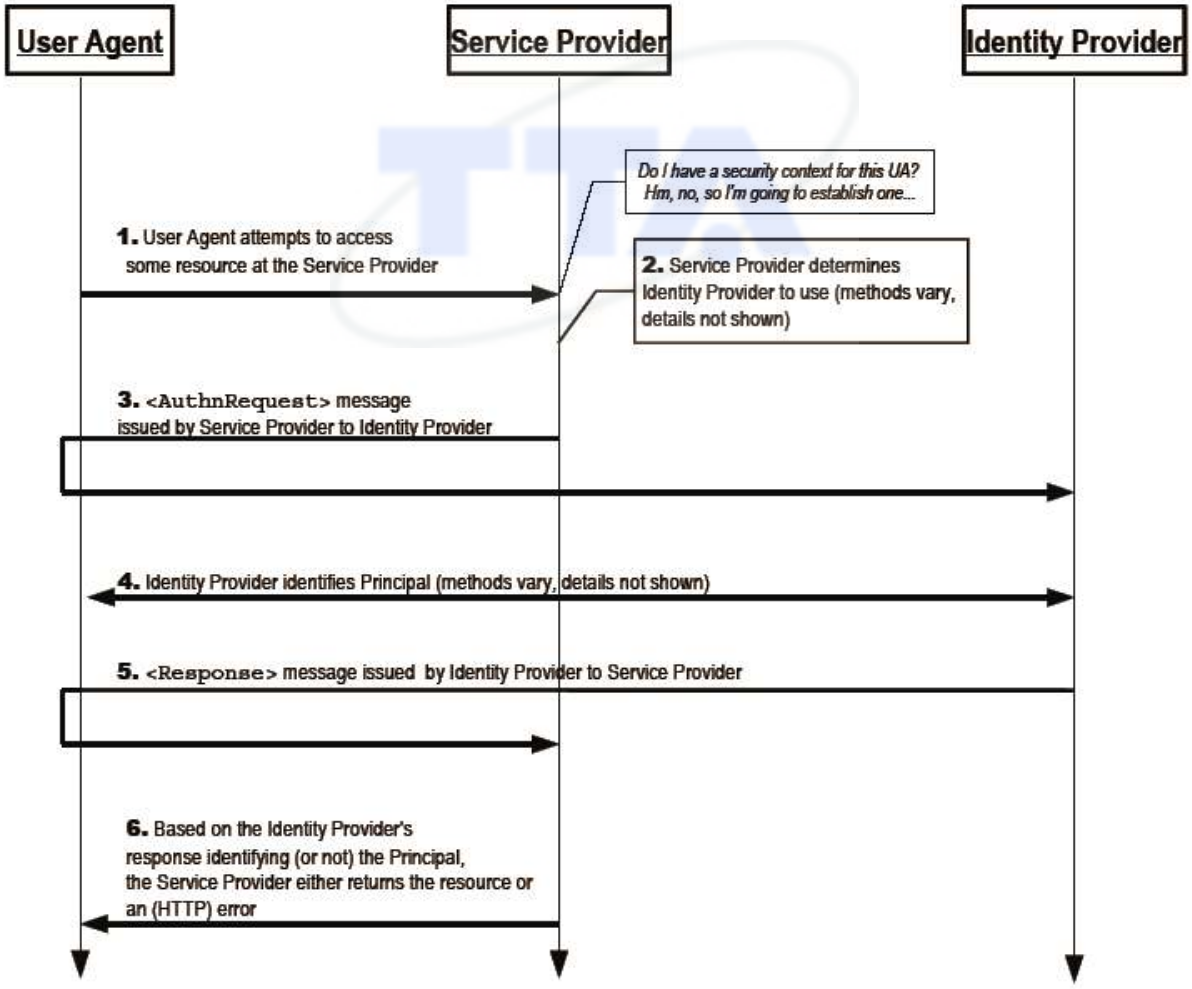
SAML Confirmation Method Identifiers: SAML V2.0 "bearer" 확인 방법 식별자인 urn:oasis:names:tc:SAML:2.0:cm:bearer가 이 프로파일에서 사용된다.

Description: 아래에 주어짐.

Updates: 없음.

5.1.2. 프로파일 개요

그림 5-1은 SSO를 달성하기 위한 기본 템플릿(template)을 도식화한 것이다. 다름 단계들은 이 프로파일에서 설명된다. 개별적인 단계 내에서, 그 단계에 대한 바인딩과 다른 구현 의존적인 행동들에 따라 하나 또는 그 이상의 실질적인 메시지 교환들이 있을 수 있다.



(그림 5-1) SSO 를 위한 기본 동작 흐름도

1. SP에 HTTP 요청

단계 1에서, 사용자는 HTTP 사용자 에이전트를 통해 보안 문맥 없이 SP에게 SP의 자원에 대한 안전한 접근 요청을 HTTP로 전달한다.

2. SP가 IdP를 결정함

단계 2에서, SP는 인증 요청 프로토콜에 대하여 자신이 선호하는 바인딩을 지원하는 IdP의 엔드포인트(endpoint) 위치를 얻는다. 이것이 달성되는 수단들은 구현에 따라 달라진다. SP는 5.3절에서 설명되는 SAML IdP 발견 프로파일을 사용할 수 있다.

3. SP가 IdP에게 <AuthnRequest>를 발급

단계 3에서, SP는 사용자 에이전트를 통해 IdP에게 배달되는 <AuthnRequest> 메시지를 발급한다. HTTP Redirect, HTTP POST 또는 HTTP Artifact 바인딩 중에 하나가 사용되어, 이 메시지가 사용자 에이전트를 통하여 IdP에게 전달되도록 할 수 있다.

4. IdP가 사용자(Principal)을 식별함

단계 4에서, IdP는 이 프로파일 영역 밖의 어떤 수단을 통해 사용자를 식별한다. 이것은 새로운 인증 동작을 요구할 수도 있고 기존의 인증 세션을 재사용할 수도 있다.

5. IdP가 <Response>를 SP에게 발급함

단계 5에서, IdP는 사용자 에이전트를 통해 SP에게 배달되는 <Response> 메시지를 발급한다. HTTP POST 또는 HTTP Artifact 바인딩 중에 하나가 사용되어, 이 메시지가 사용자 에이전트를 통하여 SP에게 전달되도록 할 수 있다. 이 메시지는 에러를 가리킬 수 있다. 만약 그렇지 않다면 최소한 하나의 인증 주장을 포함할 것이다. 응답은 일반적으로 대부분의 사용자 에이전트에서 허용하는 URL 길이를 초과할 것이기 때문에, HTTP Redirect 바인딩은 사용되지 않아야만 한다.

6. SP는 사용자의 접근을 허가하거나 또는 거절함

단계 6에서, SP가 IdP로부터 응답을 수신 받은 후에, SP는 사용자(principal)의 사용자 에이전트에게 에러 메시지를 출력할 수 있다. 만약 에러가 발생하지 않았으면, SP는 사용자에 대하여 자신의 보안 문맥을 설정할 수 있고 요청된 자원을 반환할 수 있다.

IdP가 단계 5에서 이 프로파일을 시작하여 이전 단계를 거치지 않고 SP에게 <Response>

메시지를 발급할 수 있다.

5.1.3. 프로파일 설명

만약 이 프로파일이 SP 에서 시작되면, 5.1.3.1 절부터 시작한다. 만약 IdP 에 의해서 시작되면, 5.1.3.5 절부터 시작한다. 아래 설명에서 다음 내용이 참조된다.

SSO(Single Sign-On) Service

이것은 사용자 에이전트를 통해 <AuthnRequest> 메시지나 또는 이 메시지를 나타내는 artifact가 배달되는 IdP의 인증 요청 프로토콜 엔드포인트다.

Assertion Consumer Service

이것은 사용자 에이전트를 통해 <Response> 메시지나 또는 이 메시지를 나타내는 artifact가 배달되는 SP의 인증 요청 프로토콜 엔드포인트다.

5.1.3.1. SP 에 HTTP 요청

만약 처음 접근이 SP 에 대한 것이면, 어떤 자원에 대한 임의의 요청도 이 프로파일을 시작시킬 수 있다. 요청의 형식(form)에 대한 어떠한 제약도 없다. SP 는 자신이 원하는 수단들을 자유롭게 사용해서, 최초 요청과 그 다음에 발생하는 상호작용들을 연관시킬 수 있다. 각각의 바인딩은 SP 가 프로파일 교환과 최초 요청을 연관시키는데 사용할 수 있는 RelayState 메커니즘을 제공한다. 프로파일을 사용할 때 프라이버시 조치들을 요구한다면, SP 는 RelayState 값에서 가능하면 요청과 관련된 내용들을 최소한으로 노출시켜야 한다.

5.1.3.2. SP 가 IdP 를 결정함

이 단계는 구현에 따라 달라진다. SP 는 5.3 절에서 설명되는 SAML IdP 발견 프로파일을 사용할 수 있다. SP 는 또한 적절한 IdP 를 결정할 수 있는 또 다른 서비스에게 사용자 에이전트를 리다이렉트(redirect)하기로 선택할 수 있다. 이와 같은 경우, SP 는 다음 단계에서처럼 <AuthnRequest>를 해당 SP 에게 발급하여 IdP 에게 전달되도록 할 수 있거나 또는 SP 자신을 대신하여 <AuthnRequest> 메시지를 발급하는 중개자(intermediary) 서비스를 의존할 수도 있다.

5.1.3.3. SP 가 IdP 에게 <AuthnRequest>를 발급

일단 IdP 가 선택되면, <AuthnRequest>를 송신할 때 SP 가 선택한 SAML 바인딩을 기반으로 하여, IdP 의 SSO 서비스 위치가 결정된다. 메타데이터가 이러한 목적으로 사용될 수 있다. 사용자 에이전트에 의한 HTTP 요청에 대한 응답으로, 사용되는 SAML 바인딩에 따라 <AuthnRequest> 메시지를 포함하거나 또는 artifact 를 포함하는 하나의 HTTP 응답이 반환되며, 이것은 IdP 의 SSO 서비스로 배달된다.

이 HTTP 응답과 이를 뒤따르는 SSO 서비스에게 전달되는 HTTP 요청의 정확한 포맷은 사용되는 SAML 바인딩에서 정의된다. <AuthnRequest> 메시지의 내용에 대한 프로파일에 고유한(Profile-specific) 규칙들은 5.1.4.1 절에 포함된다. 만약 HTTP Redirect 또는 POST 바인딩이 사용된다면, <AuthnRequest> 메시지가 이 단계에서 IdP 에게 직접 배달된다. 만약 HTTP Artifact 바인딩이 사용되면, 6 장에서 정의된 Artifact Resolution 프로파일이 IdP 에서 사용되며, 이것은 예를 들어 SOAP 바인딩을 사용하여, IdP 가 <AuthnRequest> 메시지를 검색하기 위하여 SP 에게 콜백(callback)을 요청한다.

이 단계에서는 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 HTTP 교환을 수행할 것이 권고된다. 만약 요청 발급자에 대한 인증이 필요하면, <AuthnRequest> 메시지는 서명될 수 있다. 만약 HTTP Artifact 바인딩이 사용되면, 이것은 또한 artifact 가 디레퍼런스(dereference)할 때, 요청 발급자를 인증하는 또 하나의 수단을 제공한다.

IdP 는 SAML 2.0 주장과 프로토콜 표준에서 설명된 것처럼 <AuthnRequest> 메시지를 처리해야만 한다. 이것은 예를 들어 만약 IsPassive 속성이 포함된다면, 사용자 에이전트와의 순차적인 상호작용을 제약할 수 있다.

5.1.3.5. IdP 가 사용자(Principal)을 식별함

이전 또는 이후 단계 중의 어느 시점에서, IdP 가 SP 에게 에러를 반환하지 않는다면, 사용자(principal)의 신원을 확인해야만 한다. 만약 ForceAuthn <AuthnRequest> 속성이 존재하면, 이것은 IdP 가 사용자에 대하여 가지고 있는 기존 세션을 의지하지 말고, 새롭게 사용자의 신원을 확인할 것을 강요한다. 만약 그렇지 않다면, IdP 는 사용자 에이전트를 인증하는데 어떠한 수단을 사용해도 된다. 이 경우,

<RequestedAuthnContext> 요소의 형식으로 <AuthnRequest>에 포함되어 있는 요구사항들에 의해 영향을 받는다.

5.1.3.6. IdP 가 <Response>를 SP 에게 발급함

<AuthnRequest>를 처리하는 것이 성공하던지 또는 실패하던지 관계 없이, IdP는 <Response> 메시지나 또는 artifact를 포함하는 HTTP 응답을 사용자 에이전트에게 제공하여 SP의 주장 소비자 서비스(assertion consumer service)에 배달되도록 해야 한다. <Response> 메시지 또는 artifact 중에 어느 것을 생성하여 전달하느냐는 사용되는 SAML 바인딩에 따라 결정된다.

이 HTTP 응답과 이를 뒤따르는 주장 소비자 서비스에 대한 HTTP 요청의 정확한 형태는 사용되는 SAML 바인딩에서 정의된다. <Response>의 내용에 대한 프로파일에 고유한 규칙들은 5.1.4.2절에 포함된다. 만약 HTTP POST 바인딩이 사용되면, <Response> 메시지는 이 단계에서 SP에게 직접적으로 배달된다. 만약 HTTP Artifact 바인딩이 사용되면, 6장에서 정의된 Artifact Resolution 프로파일이 SP에 의해 사용되며, 이것은 예를 들어 SOAP 바인딩을 사용하여, SP가 <Response> 메시지를 검색하기 위하여 IdP에게 콜백(callback)을 요청한다.

주장 소비자 서비스의 위치는 메타데이터를 사용하여 결정될 수 있다. IdP는 이 위치가 해당되는 SP에 의해 실제로 제어되고 있다는 것을 확인할 수 있는 어떠한 수단들을 가지고 있어야만 한다. SP는 어떤 SAML 바인딩과 특정 주장 소비자 서비스가 사용되는지를 그것의 <AuthnRequest>에서 지시할 수 있으며, IdP는 가능하면 이것을 존중해야만 한다.

이 단계에서는 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 HTTP 요청이 수행할 것이 권고된다. 만약 HTTP POST 바인딩이 사용되면, <Response>에 있는 <Assertion> 요소(들)은 반드시 서명되어야만 한다. 만약 HTTP Artifact 바인딩이 사용되면, <Response>에 있는 <Assertion> 요소(들)은 서명될 수 있다.

SP는 <Response> 메시지와 동봉된(enclosed) 어떠한 <Assertion> 요소들도 SAML 2.0 주장과 프로토콜 표준에서 설명된 방식으로 처리해야만 한다.

5.1.3.7. SP 는 사용자의 접근을 허가하거나 또는 거절함

이 프로파일을 완료하기 위해, SP 는 <Response>와 <Assertion>을 처리하고 자원에 대한 접근을 허가하거나 또는 거절한다. SP 는 자신이 선택한 방식의 세션 메커니즘을 사용하여 사용자 에이전트와의 보안 문맥을 설정할 수 있다. 제공된 <Assertion>을 나중에 어떻게 사용하느냐는 SP 와 다른 의지하는 측들의 판단에 따르며, 이 경우 <Assertion> 내부에 포함된 사용에 대한 제약들에 영향을 받는다.

5.1.4. Authentication Request 프로토콜의 사용

이 프로파일은 SAML 2.0 주장과 프로토콜 표준에서 정의된 Authentication Request 프로토콜을 기반으로 한다. 이 프로토콜에서, SP는 요청 발급자이며 의지하는 자(relying party)가 된다. 또한 사용자(principal)은 요청의 제출자(presenter)이며 요청되는 주체이며 확인하는 엔티티(confirming entity)가 된다. IdP의 판단에 따라 추가적으로 의지하는 자들이나 또는 확인하는 엔티티들이 존재할 수 있다.

5.1.4.1. <AuthnRequest> 사용법

SP 는 SAML 2.0 주장과 프로토콜 표준에서 설명된 어떠한 메시지 내용도 포함할 수 있다. 모든 처리 규칙은 SAML 2.0 주장과 프로토콜 표준에서 정의된 것과 같다. <Issuer> 요소는 반드시 존재해야만 하고 요청하는 SP 의 유일한 식별자를 포함해야만 한다. Format 속성은 생략되거나 또는 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 값을 가져야만 한다.

만약 IdP 가 요청을 충족시킬 수 없거나 충족시키지 않으려 한다면, IdP 는 적절한 에러 상태 코드를 포함하는 <Response> 메시지를 가지고 응답해야만 한다.

만약 사용자에 대한 식별자들이 존재하지 않을 경우, IdP 가 사용자에 대해 새로운 식별자들을 설정할 수 있도록 SP 가 허가하기를 원한다면, SP 는 AllowCreate 속성이 “true”로 설정되는 <NameIDPolicy>를 포함해야만 한다. 만약 그렇지 않다면, IdP 가 SP 에 의해서 사용될 수 있는 식별자를 이미 설정해놓은 사용자만이 성공적으로 인증될 수 있다.

SP 가 주장을 받기를 원하는 실질적인 아이덴터티(identity)를 지칭하는 <Subject> 요소를 요청에 포함할 수 있다. 이 요소는 어떠한 <SubjectConfirmation> 요소들을 포함하지 않아야만 한다. 만약 IdP 가 사용자를 이 아이텐터티로 인식하지 못하면, IdP 는 주장을 포함하지 않고 하나의 에러 상태를 포함하는 <Response> 메시지를 가지고 응답해야만 한다.

<AuthnRequest> 메시지는 사용되는 SAML 바인딩에서 지시하는 대로 서명될 수 있다. 만약 HTTP Artifact 바인딩이 사용되면, 당사자들의 인증은 선택적이며 바인딩에서 허가하는 어떠한 메커니즘도 사용될 수 있다.

만약 <AuthnRequest>가 인증되지 않고/또는 무결성이 보호되지 않으면, 이 요소에 포함된 정보는 조언하는(advisory) 것을 제외하고는 신뢰되지 않아야만 한다. 요청이 서명이 되는 것에 관계없이, IdP 는 요청에 포함된 어떠한 <AssertionConsumerServiceURL> 또는 <AssertionConsumerServiceIndex> 요소들이 SP 에 속한다는 것을 보장해야만 한다. 이 때, SP 는 IdP 에 의해 응답이 전달되는

서비스 제공자이다. 이렇게 하지 않으면, 중재자 공격(man-in-the-middle attack)을 초래할 수 있다.

<Response> 사용법

만약 IdP 가 에러를 반환하기를 원한다면, IdP 는 <Response> 메시지에 어떠한 주장들도 포함하지 않아야만 한다. 만약 그렇지 않다면, 그리고 요청이 성공적이거나 또는 응답이 요청과 관련 없이 전달되는 것이면, <Response> 요소는 다음의 규칙들을 따라야만 한다:

- <Issuer> 요소는 생략될 수 있다. 그러나 만약 존재하면 이것은 응답을 발급하는 IdP의 유일한 식별자를 포함해야만 한다; Format 속성은 생략되거나 또는 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 값을 가져야만 한다.
- 응답은 적어도 하나의 <Assertion>을 포함해야만 한다. 각각의 주장의 <Issuer> 요소는 발급하는 IdP의 유일한 식별자를 포함해야만 한다; Format 속성은 생략되거나 또는 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 값을 가져야만 한다.
- 하나 또는 그 이상의 주장들의 집합은 IdP가 사용자(principal)를 인증한 사실을 반영하도록 최소한 하나의 <AuthnStatement>를 포함해야만 한다.
- <AuthnStatement>를 포함하는 최소한 하나의 주장은 urn:oasis:names:tc:SAML:2.0:cm:bearer 방식을 포함하는 최소한 하나의 <SubjectConfirmation> 요소를 가진 하나의 <Subject> 요소를 포함해야만 한다. 만약 IdP가 5.4절에서 정의된 단일 로그아웃(Single Logout) 프로파일을 지원한다면, 이와 같은 어떠한 인증문들도 서비스 제공자가 세션당 로그아웃 요청을 할 수 있도록 SesstionIndex 속성을 포함해야만 한다.
- 위에서 설명된 운반자 <SubjectConfirmation> 요소는 SP의 주장 소비자 서비스 URL을 포함하는 하나의 Recipient 속성과 주장이 배달될 수 있는 시간 윈도우(window)를 제약하는 NotOnOrAfter 속성을 포함하는 하나의 <SubjectConfirmationData> 요소를 포함해야만 한다. 그것은 주장이 배달될 수 있는 클라이언트 주소를 제약하는 Address 속성을 포함할 수 있다. 그것은 NotBefore 속성을 포함해서는 안 된다. 만약 포함하는 메시지가 <AuthnRequest>에 대한 응답인 경우, InResponseTo 속성은 요청자의 ID와 일치해야 한다.
- 다른 문장들과 확인 방법들은 IdP의 판단에 따라 주장(들)에 포함될 수 있다. 특히, <AttributeStatement> 요소들은 포함될 수 있다. <AuthnRequest>는 SAML 메타데이타 표준에서 바람직하거나 또는 필요한 속성들에 대한 정보를 참조하는 AttributeConsumingServiceIndex XML 속성을 포함할 수 있다. IdP는 이것을 무시할 수도 있고 또는 그것의 판단에 따라 다른 속성들을 송신할 수 있다.
- 운반자 주체 확인을 포함하는 주장은 SP의 유일한 식별자를 <Audience>로 포함하는 <AudienceRestriction>을 포함해야만 한다.

- 다른 조건들 그리고 다른 <Audience> 요소들은 SP에서 요청되는 대로 또는 IdP의 판단에 따라 포함될 수 있다. 물론 이와 같은 모든 조건들은 주장이 유효한 것으로 간주되기 위해 SP에 의해 이해되고 받아들여져야만 한다. 만약 있다면, IdP가 <AuthnRequest>에 있는 요청된 <Conditions> 집합을 존중할 의무가 있지는 않다(NOT).

5.1.4.2. <Response> 메시지 처리 규칙

사용되는 SAML 바인딩에 관계없이, SP 는 다음과 같이 처리해야만 한다:

- 주장이나 또는 응답에 나타나는 어떠한 서명도 확인한다.
- 모든 운반자 <SubjectConfirmationData>에 존재하는 Recipient 속성이 <Response> 또는 artifact가 배달되는 주장 소비자 서비스 URL과 일치하는지를 확인한다.
- 모든 운반자 <SubjectConfirmationData>에 존재하는 NotOnOrAfter 속성이 현재 시간이 지난 것인지 확인한다. 이때, 제공자들 사이에 허용가능한 시각 휨(closk skew)을 적용한다.
- 운반자 <SubjectConfirmationData>에 존재하는 InResponseTo 속성이 그것의 원래의 <AuthnRequest> 메시지의 ID와 일치하는지 확인한다. 만약 응답이 요청 없이 이루어지는 자발적인 것이라면, 이 경우에는 이 속성은 존재하지 않아야만 한다.
- 의지하는 모든 주장들이 다른 면들에 있어 유효하다는 것을 확인한다.
- 만약 운반자 <SubjectConfirmationData>가 Address 속성을 포함한다면, SP는 그것과 사용자 에이전트의 클라이언트 주소를 검사할 수 있다.
- 유효하지 않는 모든 주장과 또는 그것의 주체 확인 요구사항이 충족될 수 없는 모든 주장은 폐기되어야 하며, 사용자에 대한 보안 문맥을 설정하는데 사용되지 않아야 한다.
- 만약 사용자에 대한 보안 문맥을 설정하는데 사용된 <AuthnStatement>가 SessionNotOnOrAfter 속성을 포함하면, 그리고 SP가 이 프로파일의 사용을 반복함으로써 사용자 신원을 재확인하지 않는다면, 보안 문맥은 이 속성의 시각이 도달하면 폐기되어야 한다.

5.1.4.3. Artifact 에 고유한 <Response> 메시지 처리 규칙

만약 HTTP Artifact 바인딩이 <Response>를 배달하기 위해 사용되면, Artifact Resolution 프로파일을 사용하여 artifact 를 디레퍼런스(dereference)할 때는, SP 와 IdP 는 서로 인증하여야 하며, 무결성을 보호하고 기밀성을 유지하여야만 한다.

IdP 는 <Response> 메시지가 발급된 SP 만이 <ArtifactResolve> 요청의 결과로써 메시지를 받을 수 있다는 것을 보장해야만 한다.

artifact 를 디레퍼런스(dereference)하는데 사용되는 SAML 바인딩 또는 메시지 서명 둘 중에 하나가 당사자들을 인증하고 메시지를 보호하는데 사용될 수 있다.

5.1.4.5. POST 에 고유한 처리 규칙

만약 HTTP POST 바인딩이 <Response>를 배달하는데 사용되면, 동봉된 주장은 반드시 서명이 되어야만 한다.

SP 는 어느 정도 시간 동안 사용된 ID 값들의 집합을 유지함으로써, 운반자 주장들이 재전송(replayed) 공격을 받지 않도록 보장해야만 한다. 이 기간 동안, 주장은 <SubjectConfirmationData>의 NotOnOrAfter 속성에 근거하여 유효성을 판단할 수 있을 것이다.

5.1.5. 자발적인 응답(Unsolicited Responses)

어떠한 IdP 도 SP 에게 자발적인 <Response> 메시지를 배달함으로써 이 프로파일을 시작할 수 있다.

자발적인 <Response>는 InResponseTo 속성을 포함해서도 안되며 어떠한 운반자 <SubjectConfirmationData> 요소도 이 속성을 포함하지 않아야만 한다. 만약 메타데이터가 사용되면, <Response> 또는 artifact 가 디폴트(default)로 설정된 SP 의 <md:AssertionConsumerService> 엔드포인트로 배달되어야 한다.

특히, IdP 는 바인딩에 고유한 “RelayState” 파라미터를 포함할 수 있다는 것에 주의한다. 이 파라미터는 SP 와의 상호 협약에 근거하여, 사용자 에이전트와의 이후 상호작용을 어떻게 처리할 것인지를 나타낸다. 이것은 SP 에 있는 자원의 URL 이 될 수 있다. 응답을 성공적으로 처리한 이후에, SP 가 사용자 에이전트에게 전달하는 디폴트 위치를 지시함으로써 자발적인 응답들을 처리하는 것을 준비해야 한다.

5.1.6. 메타데이터 사용

SAML 2.0 메타데이터 표준은 지원되는 바인딩들과 SP 가 이 프로파일을 이용하여 IdP 에게 요청을 전달하는 위치(들)을 설명하기 위해 하나의 엔드포인트 요소인 <md:SingleSignOnService>를 정의한다.

<md:IDPSSODescriptor> 요소의 WantAuthnRequestsSigned 속성은 IdP 가 요청이 서명되어야 한다는 요구사항을 문서화하기 위해 사용될 수 있다. <md:SPSSODescriptor> 요소의 AuthnRequestsSigned 속성은 SP 가 그것의 모든 요청을 서명하겠다는 의도를 문서화하기 위해 사용될 수 있다.

이들 제공자는 요청, 응답 그리고 주장을 서명하는데 사용되는 키(들)를 sign 을 값으로 갖는 use 속성을 가지는 <md:KeyDescriptor> 요소를 가지고 문서화할 수 있다. SAML 요소들을 암호화할 때, encrypt 을 값으로 갖는 use 속성을 가지는 <md:KeyDescriptor> 요소들이 지원되는 암호화 알고리즘들과 설명 그리고 벌크(bulk) 암호화 키들을 수신하는데 사용되는 공개키들을 문서화하는데 사용될 수 있다.

색인된 엔드포인트 요소인 <md:AssertionConsumerService>는 지원되는 바인딩과 IdP 가 이 프로파일을 사용하여 SP 에게 응답을 전달하는 SP 의 위치를 설명하는데 사용된다. index 속성은 <AuthnRequest> 메시지에서 참조를 통해 명기될 수 있는 가능한 엔드포인트들을 구별하는데 사용된다. idDefault 속성은 만약 요청에서 색인이 명기되지 않으면, 사용되는 엔드포인트를 명기하는데 사용된다.

<md:SPSSODescriptor> 요소의 WantAssertionsSigned 속성은 SP 가 이 프로파일을 가지고 개발되는 주장들이 서명되어야 한다는 요구사항을 문서화하기 위해 사용될 수 있다. 이것은 또한 특정 바인딩의 사용에 의해 부과되는 서명에 대한 요구사항외에 추가되는 사항이다. IdP 는 이렇게 할 필요는 없지만, 서명되지 않은 주장이 충분하지 않을 수 있을 가능성을 인지하고 있다는 것에 주의한다.

만약 요청 또는 응답 메시지가 HTTP Artifact 바인딩을 사용하여 배달되면, artifact 발급자는 그것의 메타데이터에 적어도 하나의 <md:ArtifactResolutionService> 엔드포인트 요소를 제공해야만 한다.

<md:IDPSSODescriptor>는 특정한 이름 식별자 포맷들, 속성 프로파일들 또는 특정 속성들과 값들을 자신이 얼마나 지원하는지를 가리키기 위해 <md:NameIDFormat>, <md:AttributeProfile> 그리고 <saml:Attribute> 요소들을 포함할 수 있다. 주어진 인증 교환 동안에 이와 같은 특징을 지원하는 능력은 정책과 IdP 의 판단에 의존한다.

<md:SPSSODescriptor> 요소는 SAML 속성들이 인증 정보와 함께 배달되는 것에 대한 SP 의 필요성이나 희망을 문서화하기 위해 또한 사용될 수 있다. 실질적인 속성들의 포함은 항상 IdP 의 판단에 따른다. 하나 또는 그 이상의 <md:AttributeConsumingService> 요소들이 그것의 메타데이터에 포함될 수 있으며, 이 때, 서로 다른 서비스를 구별하는 index 속성을 가지고 각각의 요소는 <AuthnRequest> 메시지에서 참조될 수 있다. isDefault 속성은 일단의 디폴트 속성 요구사항들을 명기하는데 사용된다.

5.2. Enhanced Client or Proxy (ECP) 프로파일

enhanced client or proxy(ECP)는 어떤 문맥에 의존적인(context-dependent) 방식으로 적절한 IdP 에 접근할 수 있는 방법을 알고 또한 SAML 2.0 바인딩 표준의 Reverse SOAP(PAOS) 바인딩을 지원하는 시스템 엔티티이다.

이 프로파일에 의해 가능한 예제 시나리오는 다음과 같다: ECP 를 사용하는 사용자(principal)가 SP 에 있는 자원을 접근하거나 또는 서비스 제공자와 원하는 자원이 이해되도록 IdP 를 접근하기 위해 그것을 사용한다. 사용자는 IdP 에서 이미 인증을 받았거나, 만약 인증을 받지 않았으면 IdP 를 방문하여 인증을 받는다. 이것은 IdP 가 인증 주장을 생성하도록 한다. 인증 주장을 생성할 때는 서비스 제공자로부터 일정 부분의 입력을 받을 수 있다. 이렇게 생성된 인증 주장은 서비스 제공자가 활용하여 웹 사용자를 위해 보안 문맥을 설정하는데 사용된다. 이러한 처리 과정 동안에, IdP 와 SP 들 간에 사용자(principal)에 대한 이름 식별자가 생성될 수 있다. 이 작업은 상호작용 파라미터들과 당사자들의 동의에 영향을 받는다.

이 프로파일은 PAOS 바인딩과 함께 SAML Authentication Request 프로토콜을 기반으로 한다.

주의: IdP 에서 사용자를 인증하는 방법은 SAML 의 범위를 넘어서는 것이다.

5.2.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp (이것은 또한 대응되는 ECP 프로파일 스키마 문서에 할당된 타깃 네임스페이스이다.)

Contact information: security-services-comment@lists.oasis-open.org

SAML Confirmation Method Identifiers: SAML V2.0 "bearer" 확인 방법 식별자 urn:oasis:names:tc:SAML:2.0:cm:bearer가 이 프로파일에서 사용된다.

Description: 아래에 주어짐.

Updates: 없음

5.2.2. 프로파일 개요

앞에서 소개된 것처럼, ECP 프로파일은 ECP 들, SP 들 그리고 IdP 들 사이의 상호작용을 명기한다. 이것은 5.1 절에서 설명된 SSO 프로파일의 특수한 응용이다. 만약 이 프로파일에서 다르게 기술되지 않는다면, 그리고 만약 브라우저에 기반한 바인딩의 사용에 특수한 규칙이 아니라면, 5.1 절에서 명기된 규칙들이 준수되어야만 한다.

ECP 는 다음 두 가지 조건들을 만족하는 클라이언트 또는 프락시이다.

- 이것은 SP와의 상호작용 문맥에서 ECP와 관련된 사용자(principal)가 사용하기를 원하는 IdP에 대한 정보를 가지고 있거나 또는 어떻게 얻는지 알고 있다.
  이것은 SP가 적절한 IdP를 알거나 또는 발견할 필요 없이 ECP에게 인증 요청을 할 수 있도록 해 준다. 이렇게 하면, 5.1절에의 SSO 프로파일의 단계 2를 효과적으로 통과하게

된다.

- 이것은 인증 요청과 응답을 위하여 여기서 프로파일링 되는 것처럼 reverse SOAP(PAOS) 바인딩을 사용할 수 있다.

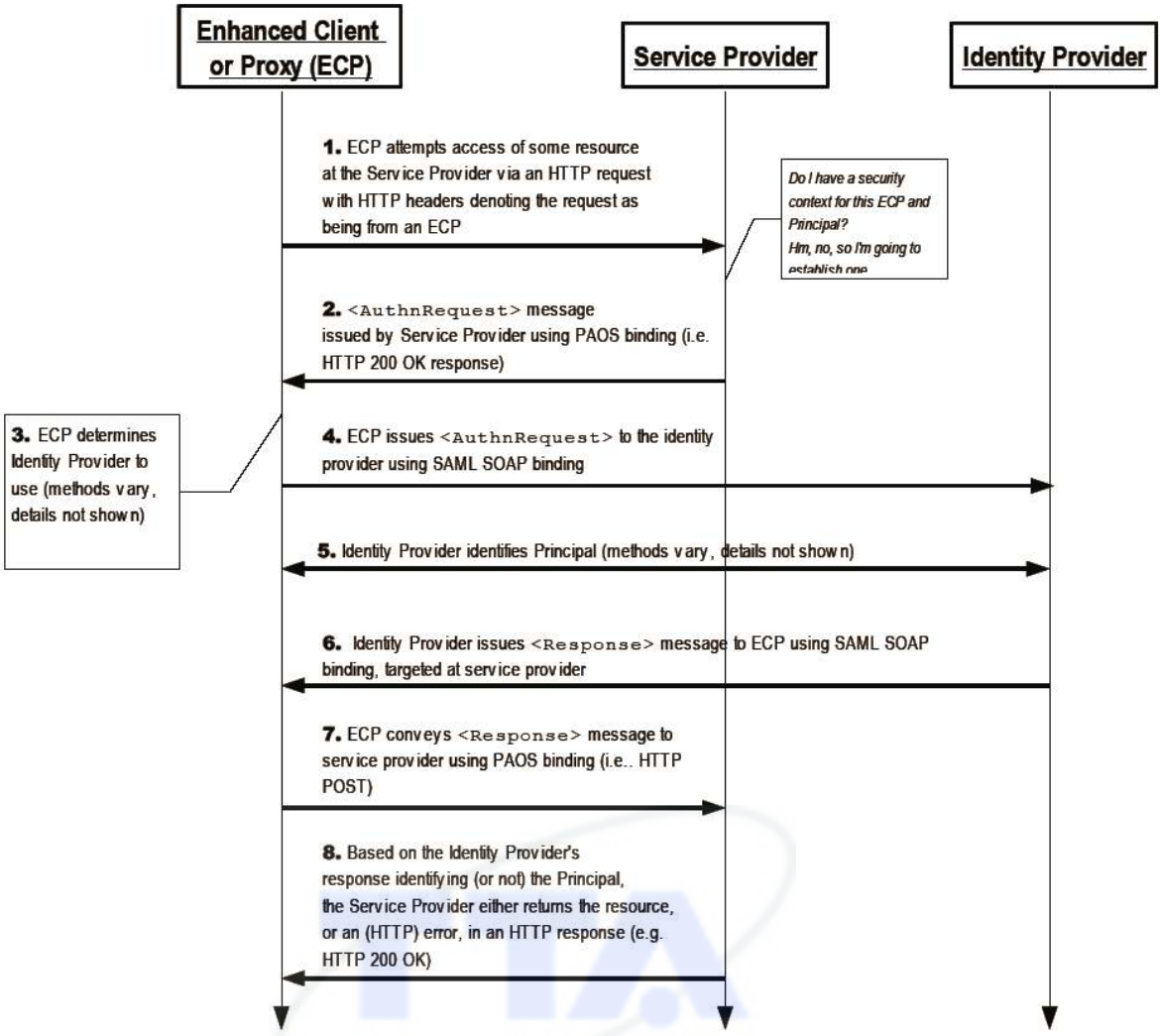
  이것은 직접적으로 주소로 접근할 수 있지도 않고 또한 항상 이용 가능하지도 않은 ECP를 통해 SP가 인증 주장을 얻을 수 있도록 해 준다. 이것은 또한 상호운용이 가능한 잘 정의된 교환 패턴과 프로파일을 사용하면서 동시에 SOAP의 장점들을 활용한다. ECP는 SP와 IdP 사이에 존재하는 하나의 SOAP 중개자(intermediary)로 볼 수 있다.

enhance client는 이 프로파일에서 설명되는 기능을 지원하는 브라우저 또는 다른 사용자 에이전트일 수 있다. enhanced proxy는 예를 들어 WAP 게이트웨이처럼, enhanced client의 동작을 모방하는(emulate) HTTP 프락시이다. 만약 다르게 언급되지 않는다면, enhanced client들을 언급하는 모든 문장들은 enhanced client들뿐만 아니라 enhanced client proxies들로 모두에 대한 문장으로써 이해되어야 한다.

enhanced client는 HTTP 요청과 응답의 몸체(body)로 메시지들을 송신하고 수신하기 때문에, 이것은 프로토콜 메시지의 크기에 어떠한 제약도 없다.

이 프로파일은 Reverse SOAP(PAOS) 바인딩을 활용한다. 이 프로파일의 구현자들은 PAOS 바인딩에서 명기된 PAOS 지원을 가리키는 HTTP 표시에 대한 규칙들을 따라야만 하며, 또한 이 프로파일에서 명기된 규칙들도 따라야만 한다. 이 프로파일은 HTTP 응답자와 ECP 사이에서 운반되는 PAOS SOAP 헤더 블록을 활용한다. 그러나 PAOS 자체를 정의하지는 않는다. 이 프로파일은 SAML 요청과 응답에 수반되는 SOAP 헤더 블록들을 정의한다. 이 헤더 블록들은 필요에 따라 다른 SOAP 헤더 블록들로 만들 수 있다. 예를 들어, 만약 전자 서명이 인증 요청에 적용되어야 한다면, 필요한 보안 기능을 추가하기 위해 SOAP 메시지 보안 헤더 블록이 사용될 수 있다.

두 개의 요청/응답 SOAP 헤더 블록들이 사용된다: 일반적인 PAOS 정보를 위해서는 PAOS 헤더 블록과 ECP 프로파일 기능에 특정한 정보를 운반하는 ECP 프로파일에 고유한 헤더 블록들.



(그림 5-2) ECP 프로파일 흐름도

(그림 5-2)는 ECP를 사용하여 SSO를 제공하기 위한 기본 템플릿을 도식화한다. 다음 단계들은 이 프로파일에서 설명된다. 개별적인 단계 내에서, 그 단계에 대한 바인딩과 다른 구현 의존적인 행동들에 따라 하나 또는 그 이상의 실질적인 메시지 교환들이 있을 수 있다.

1. ECP가 SP에게 HTTP 요청

   단계 1에서, ECP를 통해 사용자는 ECP와 사용자에 대한 보안 문맥을 설정해 놓지 않은 SP에게 SP의 자원에 대한 안전한 접근 요청을 HTTP로 전달한다.

2. SP가 ECP에게 <AuthnRequest> 발급

   단계 2에서, SP는 ECP에게 <AuthnRequest> 메시지를 발급하며, 이 메시지는 ECP를 통해 적절한 IdP에게 배달되도록 한다. Reverse SOAP(PAOS) 바인딩이 여기에서 사용된다.

3. ECP가 IdP를 결정

   단계 3에서, ECP는 인증 요청 프로토콜에 대하여 자신이 선호하는 바인딩을 지원하는 IdP의 엔드포인트의 위치를 얻는다. 이것이 달성되는 수단은 구현에 의존적이다. ECP는 5.3절에서 설명되는 SAML IdP 발견 프로파일을 사용할 수 있다.

4. ECP가 <AuthnRequest>를 IdP에게 전달

단계 4에서, ECP는 단계 3에서 식별된 IdP에게 <AuthnRequest>를 운반한다. 이 때, IdP가 SAML 요청에 응답하기 전에 ECP와 임의의 HTTP 메시지를 교환하는 것을 추가적으로 허용하도록 하는 변형된 SAML SOAP 바인딩을 사용한다.

5. IdP가 사용자를 식별

단계 5에서, IdP는 이 프로파일 영역 밖의 어떤 수단을 이용하여 사용자를 식별한다. 이것은 새로운 인증 동작을 요구할 수도 있고 기존의 인증 세션을 재사용할 수도 있다.

6. IdP가 SP를 목적으로 하는 <Response>를 ECP에게 발급

단계 6에서, IdP는 SAML SOAP 바인딩을 사용하여 <Response> 메시지를 발급하며, 이 메시지는 ECP를 통해 SP에게 배달될 것이다. 이 메시지는 에러를 가리킬 수 있다. 만약 그렇지 않다면 최소한 하나의 인증 주장을 포함할 것이다.

7. ECP가 SP에게 <Response> 메시지를 전달

단계 7에서, ECP를 PAOS 바인딩을 사용하여 SP에게 <Response> 메시지를 운반한다.

8. SP가 사용자의 접근을 허가하거나 거절함

단계 8에서, SP가 IdP로부터 <Response> 메시지를 수신 받은 후에, SP는 사용자(principal)의 사용자 에이전트에게 에러 메시지를 출력할 수 있다. 만약 에러가 발생하지 않았으면, SP는 사용자에 대하여 자신의 보안 문맥을 설정하고 요청된 자원을 반환한다.

5.2.3. 프로파일 설명

다음 절에서 개별적인 단계들을 자세히 정의한다.

5.2.3.1. ECP가 SP에게 HTTP 요청

ECP 는 접근하려는 자원을 명기하면서, SP 에게 HTTP 요청을 전달한다. 이 HTTP 요청은 PAOS 바인딩을 따라야만 하며, 이것은 HTTP 요청이 반드시 다음 HTTP 헤더 필드들을 포함해야만 한다는 것을 의미한다:

1. MIME 타입 “application/vnd.paos+xml”을 받아들일 수 있는 능력을 가리키는 HTTP Accept 헤더
2. 최소한 “urn:liberty:paos:2003-08”을 가지는 PAOS 버전을 명기하는 HTTP PAOS 헤더 필드
3. 또한, 이 프로파일의 지원은 하나의 서비스 값으로 HTTP PAOS 헤더 필드에 명기되어야만 한다. 이 때, 이 값은 urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp이 된다. 이 값은 PAOS 요청 SOAP 헤더 블록의 서비스 속성과 대응되어야 한다.

예를 들어, 사용자 에이전트는 다음과 같이 서비스 요청자로부터 어떠한 페이지를 요청할 수

있다.

```
GET /index HTTP/1.1
Host: identity-service.example.com
Accept: text/html; application/vnd.paos+xml
PAOS: ver='urn:liberty:paos:2003-08' ;
'urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp'
```

5.2.3.2. SP 가 ECP 에게 <AuthnRequest> 발급

SP가 명기된 자원의 접근을 허용하기 전에, 즉 서비스나 데이터를 제공하기 전에, SP가 사용자에 대한 보안 문맥을 요구할 때, SP는 PAOS 바인딩을 사용하여, HTTP 응답에 <AuthnRequest> 메시지를 포함함으로써 HTTP 요청에 응답할 수 있다. SP는 단일한 SOAP 봉투(envelope)를 포함하는 HTTP 200 OK 응답을 ECP에게 발급할 것이다.

SOAP 봉투는 다음을 포함해야만 한다:

1. SOAP 몸체에 최종(ultimate) SOAP 수신자인 IdP를 목적으로 하는 <AuthnRequest> 요소
2. 값을 http://schemas.xmlsoap.org/soap/actor/next로 설정한 SOAP actor를 사용하여, ECP를 타깃으로 하는 PAOS SOAP 헤더 블록. 이 헤더 블록은 이 자발적-응답(solicit-response) 메시지 교환에서 응답이 보내지는 URL과 같은 제어 정보를 제공한다.
3. 값을 http://schemas.xmlsoap.org/soap/actor/next로 설정한 SOAP actor를 사용하여, ECP를 타깃으로 하는 ECP 프로파일에 고유한 Request SOAP 헤더 블록. ECP Request 헤더 블록은 ECP가 그것을 처리할 때 필요한 인증 요청과 관련된 정보를 정의한다. 이러한 정보에는 SP가 받아들일 수 있는 IdP 리스트, ECP가 클라이언트를 통해 사용자와 상호작용을 할 수 있는지 여부 그리고 사용자 화면에 출력될 수 있는 SP의 가독성 있는 이름 등이 포함된다.

SOAP 봉투는 http://schemas.xmlsoap.org/soap/actor/next를 값으로 하는 SOAP actor를 사용하여, ECP를 타깃으로 하는 ECP RelayState SOAP 헤더 블록을 포함할 수 있다. 이 헤더는 SAML 응답과 함께 ECP가 반환하는 상태 정보를 포함한다.

5.2.3.3. ECP가 IdP를 결정

ECP는 어떤 IdP가 적절하며 SOAP 메시지를 적절하게 라우트 할 것인지를 결정할 것이다.

5.2.3.4. ECP 가 <AuthnRequest>를 IdP 에게 전달

ECP는 변형된 SAML SOAP 바인딩을 사용하여, <AuthnRequest> 메시지를 IdP에게 전달하기 전에, PAOS, ECP RelayState 그리고 ECP Request 헤더 블록을 제거해야만 한다. SAML 요청은 일반적인 방식으로 SOAP을 통해 제출된다. 그러나 IdP는 예를 들어 HTML 로그인 폼 또는 다른 표현-지향적인(presentation-oriented) 응답을 포함하는 HTTP 응답을 가지고 ECP의 HTTP 요청에 응답할 수 있다. 일련의 HTTP 교환이 발생될 수 있다 그러나 최종적으로 IdP는 SAML SOAP 교환을 완료하고 SOAP 바인딩을 통해 SAML 응답을 반환해야만 한다.

<AuthnRequest> 요소는 그 자체가 SP에 의해 서명될 수 있다. 이점과 다른 점에서, 5.1.4.1절 브라우저 SSO 프로파일에서 명기된 메시지 규칙을 반드시 따라야만 한다.

이 단계의 전 또는 이후 단계에서, IdP는 어떤 수단을 이용하여 사용자의 신원을 확인해야만 한다. 만약 그렇지 않다면, IdP는 아래 5.2.3.6절에서 설명되는 것처럼, 에러 <Response>를 반환해야만 한다.

5.2.3.5. IdP가 사용자를 식별

이전 또는 이후 단계 중의 어느 시점에서, IdP 가 SP 에게 에러를 반환하지 않는다면, 사용자(principal)의 신원을 확인해야만 한다. ForceAuthn <AuthnRequest> 속성이 만약 존재하면, 이것은 IdP 가 사용자에 대하여 가지고 있는 기존 세션을 의지하지 말고, 새롭게 사용자를 식별할 것을 강요한다. 만약 그렇지 않다면, IdP 는 사용자 에이전트를 인증하는데 어떠한 수단을 사용해도 된다. 이 경우, <RequestedAuthnContext> 요소의 형식으로 <AuthnRequest>에 포함되어 있는 요구사항들에 의해 영향을 받는다.

5.2.3.6. IdP가 SP를 목적으로 하는 <Response>를 ECP에게 발급

IdP 는 사용자의 신원을 확인한 후에 SAML <Response> 메시지 또는 SOAP fault 를 반환한다. SAML SOAP 바인딩을 이용하여 SAML 응답은 SOAP 몸체에, SP 를 최종적인 SOAP 수신자로 의도하는 <Response> 요소를 가지는 SOAP 메시지 형태로 운반된다. 5.1.4.2 절 브라우저 SSO 프로파일에서 명기된 응답에 대한 규칙들이 지켜져야만 한다.

IdP 의 응답 메시지는 ECP 를 목적으로 하는 프로파일에 고유한 ECP 응답 SOAP 헤더 블록을 포함해야만 하고, 또한 ECP 를 목적으로 하는 ECP RelayState 헤더 블록을포함할 수 있다.

5.2.3.7. ECP가 SP에게 <Response> 메시지를 전달

ECP는 헤더 블록들을 제거한다. 그리고 PAOS 바인딩을 사용하여 SP에게 SOAP 응답을

전달하기 전에 PAOS Response SOAP 헤더 블록과 ECP RelayState 헤더 블록을 추가할 수 있다.

SP에게 전달되는 응답의 <paos:Response> 헤더 블록은 SP로부터 받은 요청과 이 응답을 상호 연관시키는데(correlated) 일반적으로 사용된다. 이 프로파일에서는 SAML <Response> 요소의 InResponseTo 속성이 이러한 목적으로 사용될 수 있기 때문에, 상호연관 refToMessageID 속성은 필요하지 않다. 그러나 만약 <paos:Request> SOAP 헤더 블록이 messageID를 가지고 있었다면, 반드시 <paos:Request> SOAP 헤더 블록이 사용되어야만 한다.

<ecp:RelayState> 헤더 블록은 전형적으로 ECP에게 자신의 요청을 전달하는 SP에 의해 제공된다. 그러나 만약 IdP가 대응되는 SAML 요청을 받지 않고 자발적인 응답을 생성하고 있다면, 이 응답은 SP와의 상호협약에 근거하여, ECP와 후속 상호작용을 어떻게 하는지를 가리키는 RelayState 헤더 블록을 포함할 수 있다. 이것은 SP에 있는 자원의 URL이 될 수 있다.

만약 SP가 ECP에 전달한 자신의 요청에 <ecp:RelayState> SOAP 헤더 블록을 포함했다면, 또는 만약 IdP가 자신의 응답에 <ecp:RelayState> SOAP 헤더 블록을 포함했다면, ECP는 SP에게 송신되는 SAML 응답에 동일한 헤더 블록을 포함시켜야만 한다. 만약 있다면, 이 헤더 블록에 있는 SP 값이 높은 우선 순위를 가져야만 한다.

5.2.3.8. SP가 사용자의 접근을 허가하거나 거절함

일단 SP가 PAOS를 사용하여 SOAP 봉투로 HTTP 요청 형식으로 SAML 응답을 수신하면, SP는 HTTP 응답으로 서비스 데이터를 가지고 응답할 수 있다. 응답을 소비할 때, 5.1.4.3절과 5.1.4.5절의 브라우저 SSO 프로파일에서 명기된 규칙들이 지켜져야만 한다. 즉, HTTP POST 바인딩을 가지고 <Response>를 수신할 때 사용된 것과 동일한 처리 규칙들이 PAOS 사용시에도 적용된다.

5.2.4. ECP 프로파일 스키마 사용법

ECP 프로파일 XML 스키마는 이 프로파일에서 사용된 SOAP Request/Response 헤더 블록들을 정의한다. 다음은 이 스키마 문서의 완전한 리스팅이다.

```
<schema
        targetNamespace="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
        xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:ecp="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
        xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
        xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"
        elementFormDefault="unqualified"
```

```
        attributeFormDefault="unqualified"
        blockDefault="substitution"
        version="2.0">
    <import namespace="urn:oasis:names:tc:SAML:2.0:protocol"
            schemaLocation="saml-schema-protocol-2.0.xsd"/>
    <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
            schemaLocation="saml-schema-assertion-2.0.xsd"/>
    <import namespace="http://schemas.xmlsoap.org/soap/envelope/"
            schemaLocation="http://schemas.xmlsoap.org/soap/envelope/"/>
    <annotation>
        <documentation>
            Document identifier: saml-schema-ecp-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V2.0 (March, 2005):
            Custom schema for ECP profile, first published in SAML 2.0.
        </documentation>
    </annotation>
    <element name="Request" type="ecp:RequestType"/>
    <complexType name="RequestType">
        <sequence>
            <element ref="saml:Issuer"/>
            <element ref="samlp:IDPList" minOccurs="0"/>
        </sequence>
        <attribute ref="S:mustUnderstand" use="required"/>
        <attribute ref="S:actor" use="required"/>
        <attribute name="ProviderName" type="string" use="optional"/>
        <attribute name="IsPassive" type="boolean" use="optional"/>
    </complexType>
    <element name="Response" type="ecp:ResponseType"/>
    <complexType name="ResponseType">
        <attribute ref="S:mustUnderstand" use="required"/>
        <attribute ref="S:actor" use="required"/>
        <attribute name="AssertionConsumerServiceURL" type="anyURI"
                   use="required"/>
    </complexType>
    <element name="RelayState" type="ecp:RelayStateType"/>
```

```
    <complexType name="RelayStateType">
        <simpleContent>
            <extension base="string">
                <attribute ref="S:mustUnderstand" use="required"/>
                <attribute ref="S:actor" use="required"/>
            </extension>
        </simpleContent>
    </complexType>
</schema>
```

다음 절은 이들 XML 구조들이 어떻게 사용되는지를 설명한다.

5.2.4.1. PAOS Request 헤더 블록: SP to ECP

PAOS Request 헤더 블록은 PAOS 처리의 사용을 알리며 다음 속성들을 포함한다:

responseConsumerURL [Required]

ECP가 어디로 에러 응답을 전달해야 하는지를 명기한다. 또한 ECP 응답 헤더 블록에 있는 AssertionServiceConsumerURL과 이 위치를 상호 검사함으로써, IdP의 응답의 정확성을 확인하는데 사용된다. 이 값은 <AuthnRequest>에서 운반되는 AssertionServiceConsumerURL 또는 메타데이터에서 참조되는 URL과 동일한 값이어야만 한다.

service [Required]

사용 중인 PAOS 서비스가 이 SAML 인증 프로파일임을 가리킨다. 이 값은 “urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp”이어야만 한다.

SOAP-ENV:mustUnderstand [Required]

이 값은 1(true)이어야만 한다. 만약 PAOS 헤더 블록이 이해되지 않는다면, SOAP fault가 생성되어야만 한다.

SOAP-ENV:actor [Required]

이 값은 “http://schemas.xmlsoap.org/soap/actor/next”으로 설정되어야만 한다.

messageID [Optional]

선택적인 응답 상호연관을 허용한다. 이 속성은 이 프로파일에서 사용될 수 있다. 그러나 <AuthnRequest>의 ID 속성과 <Response>의 InResponseTo 속성을 통해 상호연관 기능이 SAML 프로토콜 레이어에서 제공되기 때문에 필수요소는 아니다(NOT).

PAOS Request SOAP 헤더 블록은 어떠한 요소 내용도 가지고 있지 않다.

5.2.4.2. ECP Request 헤더 블록: SP to ECP

ECP Request SOAP 헤더 블록은 인증 요청을 처리하기 위해 ECP에서 필요한 정보를 운반하기 위해 사용된다. 이 헤더 블록은 필수요소이며, 이것의 존재는 이 프로파일을 사용한다는 것을 나타낸다. 이 헤더 블록은 다음 요소들과 속성들을 포함한다:

SOAP-ENV:mustUnderstand [Required]

이 값은 1(true)이어야만 한다. 만약 ECP 헤더 블록이 이해되지 않는다면, SOAP fault가 생성되어야만 한다.

SOAP-ENV:actor [Required]

이 값은 "http://schemas.xmlsoap.org/soap/actor/next"으로 설정되어야만 한다.

ProviderName [Optional]

요청하는 SP에 대한 사람이 읽을 수 있는 이름

IsPassive [Optional]

부울린 값. 만약 true이면, IdP와 클라이언트 자체는 요청 발급자로부터 사용자 인터페이스를 제어해서는 안되며 눈에 띄게 사용자와 상호작용을 해서는 안 된다. 만약 이 값이 제공되지 않으면, 디폴트는 true이다.

<saml:Issuer> [Required]

이 요소는 요청하는 SP의 유일한 식별자를 포함해야만 한다; Format 속성은 생략되거나, 그렇지 않다면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity을 값으로 가져야만 한다.

<samlp:IDPList> [Optional]

SP가 인지하고 ECP 그것들 중에서 요청을 서비스하기 위해 선택할 수 있는 선택적인 IdP들의 리스트. 이 요소에 대한 자세한 사항은 SAML 2.0 주장과 프로토콜 표준을 참조한다.

5.2.4.3. ECP RelayState 헤더 블록: SP to ECP

ECP RelayState SOAP 헤더 블록은 나중에 ECP 부터 받은 응답을 처리할 때 필요하게 될 상태정보를 SP 로부터 운반하는데 사용된다. 이것은 선택적인 요소이지만, 만약 사용된다면, ECP 는 단계 5 에서 응답에 동일한 헤더 블록을 포함시켜야만 한다. 이 헤더 블록은 다음 속성들을 포함한다:

SOAP-ENV:mustUnderstand [Required]

이 값은 1(true)이어야만 한다. 만약 이 헤더 블록이 이해되지 않는다면, SOAP fault가 생성되어야만 한다.

SOAP-ENV:actor [Required]

이 값은 "http://schemas.xmlsoap.org/soap/actor/next"으로 설정되어야만 한다.

이 헤더 요소의 내용은 요청자에 의해 생성된 상태 정보를 포함하는 문자열이다. 만약

제공된다면, ECP는 단계 5에서 SP에게 응답할 때, RelayState 헤더 블록에 동일한 값을 포함시켜야만 한다. 문자열은 길이가 80 바이트를 초과하지 않아야만 한다. 또한 문자열은 메시지를 전달할 때 존재하거나 또는 존재하지 않을 지도 모르는 다른 어떠한 보호 메커니즘에 의지하지 않고 요청자에 의해 무결성이 보호되어야 한다.

다음은 SP로부터 ECP에게 전달되는 SOAP 인증 요청의 예이다.

```
<SOAP-ENV:Envelope
        xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
        xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header>
        <paos:Request
            xmlns:paos="urn:liberty:paos:2003-08"
            responseConsumerURL="http://identity-service.example.com/abc"
            messageID="6c3a4f8b9c2d" SOAPENV:
            actor="http://schemas.xmlsoap.org/soap/actor/next" SOAPENV:
            mustUnderstand="1"
            service="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp">
        </paos:Request>
        <ecp:Request
                xmlns:ecp="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
                SOAP-ENV:mustUnderstand="1"
                SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next"
                ProviderName="Service Provider X" IsPassive="0">
            <saml:Issuer>https://ServiceProvider.example.com</saml:Issuer>
            <samlp:IDPList>
                <samlp:IDPEntry
                        ProviderID="https://IdentityProvider.example.com"
                        Name="Identity Provider X"
                        Loc="https://IdentityProvider.example.com/saml2/sso"
                </samlp:IDPEntry>
                <samlp:GetComplete>
                    https://ServiceProvider.example.com/idplist?id=604be136-
fe91-441e-afb8
                </samlp:GetComplete>
            </samlp:IDPList>
        </ecp:Request>
        <ecp:RelayState
```

```
                xmlns:ecp="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
                SOAP-ENV:mustUnderstand="1"
                SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next">
            ...
        </ecp:RelayState>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <samlp:AuthnRequest> ... </samlp:AuthnRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

위에서 기술하였듯이, PAOS 와 ECP 헤더 블록들은 인증 요청이 IdP 에게 전달되기 전에, ECP 에 의해 SOAP 메시지로부터 제거된다. ECP 로부터 IdP 에게 전달되는 인증 요청의 예는 다음과 같다:

```
<SOAP-ENV:Envelope
        xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    <SOAP-ENV:Body>
        <samlp:AuthnRequest> ... </samlp:AuthnRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>.
```

5.2.4.4. ECP Response 헤더 블록: IdP to ECP

ECP Response SOAP 헤더 블록은 IdP가 ECP에게 응답을 하는데 사용되어야만 한다. 이 헤더 블록은 다음 요소들과 속성들을 포함한다:

SOAP-ENV:mustUnderstand [Required]

이 값은 1(true)이어야만 한다. 만약 ECP 헤더 블록이 이해되지 않는다면, SOAP fault가 생성되어야만 한다.

SOAP-ENV:actor [Required]

이 값은 “http://schemas.xmlsoap.org/soap/actor/next”으로 설정되어야만 한다.

AssertionConsumerServiceURL [Required]

IdP가 <AuthnRequest> 메시지를 근거로 하거나 또는 IdP에 의해 얻어진 SP의 메타데이터를 근거로 하여 설정된다.

ECP는 이 값이 ECP가 SP로부터 이전에 수신했던 PAOS Request SOAP 헤더 블록에 있는 responseConsumerURL에서 얻었던 값과 일치하는지 확인해야만 한다. responseConsumerURL는

상대적(relative)일 수 있고 AssertionConsumerServiceURL은 절대적(absolute)이기 때문에, 일부 처리/정규화가 필요할 수 있다.

이 메커니즘은 응답의 올바른 목적지를 확인하기 위한 보안 목적으로 사용된다. 만약 이 값이 일치하지 않는다면, ECP는 SP에게 SOAP fault 응답을 생성해야만 하며 SAML 응답을 반환하지 않아야만 한다.

ECP Response SOAP 헤더는 어떠한 요소 내용도 가지지 않는다.

다음은 IdP-to-ECP 응답의 한 예이다.

```
<SOAP-ENV:Envelope
        xmlns:ecp="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
        xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
        xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header>
        <ecp:Response
                SOAP-ENV:mustUnderstand="1"
                SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next"
    AssertionConsumerServiceURL="https://ServiceProvider.example.com/ecp_assertion
_consumer"/>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <samlp:Response> ... </samlp:Response>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

5.2.4.5. PAOS Response 헤더 블록: ECP to SP

PAOS Response 헤더 블록은 다음 속성들을 포함한다:

SOAP-ENV:mustUnderstand [Required]

이 값은 1(true)이어야만 한다. 만약 PAOS 헤더 블록이 이해되지 않는다면, SOAP fault가 생성되어야만 한다.

SOAP-ENV:actor [Required]

이 값은 “http://schemas.xmlsoap.org/soap/actor/next”으로 설정되어야만 한다.

refToMessageID [Optional]

PAOS 요청과의 상호연관을 허용한다. 이 선택적인 속성과 헤더 블록 전체는 만약 대응되는 PAOS 요청이 messageID 속성을 기술했다면, ECP에 의해 추가되어야만 한다. 동등한 기능이 <AuthnRequest>와 <Response> 상호연관을 사용하여 SAML에서 제공된다는 것에 주의한다.

PAOS Response SOAP 헤더는 어떠한 요소 내용도 가지지 않는다.

다음은 ECP-to-SP 응답의 한 예이다.

```
<SOAP-ENV:Envelope
        xmlns:paos="urn:liberty:paos:2003-08"
        xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
        xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header>
        <paos:Response
                refToMessageID="6c3a4f8b9c2d"
                SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next/"
                SOAP-ENV:mustUnderstand="1"/>
        <ecp:RelayState
                xmlns:ecp="urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp"
                SOAP-ENV:mustUnderstand="1"
                SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next">
            ...
        </ecp:RelayState>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <samlp:Response> ... </samlp:Response>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

5.2.5. 보안 고려사항

<AuthnRequest> 메시지는 서명되어야 한다. 브라우저 SSO 프로파일에서 기술된 규칙에 따라, <Response>에 동봉된 주장들은 반드시 서명되어야만 한다. PAOS를 통하여 SOAP 봉투에 응답을 배달하는 것은 본질적으로 HTTP POST 바인딩과 유사하며, 따라서 HTTP POST 바인딩에 적절한 보안 대처 조치들이 사용된다.

SOAP 헤더들은 SOAP 메시지 보안에서와 같이 또는 클라이언트와의 모든 HTTP 교환에 대하여 TLS를 사용하는 것과 같은 방식으로 무결성이 보호되어야 한다.

SP는 예를 들어 서비스-사이드(server-side) TLS 인증을 가지고, ECP에 인증되어야 한다.

ECP는 인증된 세션을 유지하는 방식 등으로 IdP에 인증되어야 한다. <AuthnRequest> 메시지의 배달 이후부터 IdP가 <Response>를 반환하기 전에 발생하는 모든 HTTP 교환들은 안전하게 원래의 요청과 연관되어야만 한다.

5.3. Identity Provider Discovery 프로파일

이 절은 SP가 사용자가 Web Browser SSO 프로파일에서 사용 중인 IdP들을 반영할 수 있도록 하는 프로파일을 정의한다. 하나 이상의 IdP를 가지는 배치 환경에서, SP들은 사용자가 어떠한 IdP(들)을 사용하는지 발견하는 수단이 필요하다. 이 발견 프로파일은 배치 환경에서 IdP들과 SP들 사이에 공통적인 어떠한 도메인에서 쓰여진 쿠키에 의존한다. 배치시 미리 결정하는 도메인을 이 프로파일에서는 공통 도메인(common domain) 이라고 부르며, IdP들의 리스트를 포함하는 쿠키를 공통 도메인 쿠키로 불린다.

어떤 엔티티들인 공통 도메인에서 웹 서버를 호스팅할 지는 배치와 관련된 이슈이며 이 프로파일을 벗어난 문제이다.

5.3.1. Common Domain 쿠키

쿠키의 이름은 반드시 "_saml_idp"이어야만 한다. 쿠키 값의 포맷은 단일한 스페이스 문자(space character)로 분리된 하나 또는 그 이상의 base-64 인코드된 URI 값들의 집합이어야만 한다. 각각의 URI는 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼 하나의 IdP의 유일한 식별자이다. 이 값들의 최종 집합은 URL 인코드된다.

공통 보안 쿠키를 쓰는 서비스는 IdP의 유일한 식별자를 이 리스트에 추가(append)해야 한다. 만약 식별자가 리스트에 이미 존재하면, 이 식별자는 제거되고 마지막에 추가될 수 있다. 이것은 리스트의 마지막이 가장 최근에 설정된 IdP 세션이 되도록 하기 위한 것이다.

쿠키는 “/”을 Path 접두사로 설정해야만 한다. Domain은 “.[common-domain]”으로 설정되어야만 한다. 이 곳에서 [common-domain]은 이 프로파일을 사용하는 배치 환경 내에서 설정되는 공통 도메인이다.

쿠키에는 선도하는(leading) 기간이 있어야만 한다. 쿠키는 보안(secure)으로 표시되어야만 한다.

쿠키 문법은 IETF RFC 2965에 부합되어야 한다. 쿠키는 세션 동안만 존재(session-only)하거나 또는 영속적(persistent)이거나 둘 중에 하나로 설정될 수 있다. 이것은 배치 환경에서 선택될 수 있지만 배치 환경 내에 있는 모든 IdP들에게 균등하게 적용되어야 한다.

5.3.2. Common Domain 쿠키 설정

IdP가 사용자를 인증한 후에, IdP는 공통 도메인 쿠키를 설정할 수 있다. 쿠키가 위에서 주어진 파라미터들로 성공적으로 설정될 수 있다면, IdP가 쿠키를 설정하는 수단은 구현에 고유하다(implementation-specific). 하나의 가능한 구현 전략이 다음과 같이 설명되는데 이것은 비 규범적인 것으로 간주되어야 한다. IdP는 다음과 같이 할 수 있다:

- 공통 도메인에서 자신을 위한 DNS와 IP 별칭(alias)를 미리 설정해 놓는다.
- “https”를 명기하는 하나의 URL을 사용하는 DNS 별칭을 사용하여 사용자 에이전트를 그

자신에게 리다이렉트시킨다. URL의 구조는 구현에 따라 결정되는 것이며 사용자 에이전트를 식별하는데 필요한 세션 정보를 포함할 수 있다.

- 위에서 기술한 파라미터를 사용하여 리다이렉트된 사용자 에이전트에 쿠키를 설정한다.
- 사용자 에이전트를 자신에게 리다이렉트시키거나 또는 적절하다면, SP에게 리다이렉트시킨다.

5.3.3. Common Domain 쿠키 얻기

SP가 사용자가 이용하는 IdP들을 발견할 필요가 있을 때에는, SP는 공통 도메인에 있는 하나의 HTTP 서버에서 읽혀진 후에, SP에게 공통 도메인 쿠키들 제출하도록 고안된 통신 교환을 기동시킨다.

만약 공통 도메인에 있는 HTTP 서버가 SP에 의해 운영된다면 또는 만약 다른 장치들이 적절하게 있다면, SP는 최적화된 SSO 처리를 위해 공통 도메인에 있는 HTTP 서버를 활용하여 자신의 <AuthnRequest>를 IdP에게 전달할 수 있다.

SP가 사용자 에이전트가 5.3.1절에서 주어진 파라미터들로 설정된 쿠키를 제출하도록 할 수 있는 한, SP가 쿠키를 읽는 특정 방법은 구현에 고유한 것이다(implementation-specific). 하나의 가능한 구현 전략이 다음에 설명되는데 이것은 비 규범적인 것으로 간주되어야 한다. 또한, 다른 응용들을 위해서 약간의 최적화가 필요할 지도 모른다.

- 공통 도메인에서 자신을 위한 DNS와 IP 별칭(alias)를 미리 설정해 놓는다.
- “https”를 명기하는 하나의 URL을 사용하는 DNS 별칭을 사용하여 사용자 에이전트를 그 자신에게 리다이렉트시킨다. URL의 구조는 구현에 따라 결정되는 것이며 사용자 에이전트를 식별하는데 필요한 세션 정보를 포함할 수 있다.
- 사용자 에이전트를 자신에게 리다이렉트시키거나 또는 적절하다면, IdP에게 리다이렉트시킨다.

5.4. Single Logout 프로파일

일단 사용자가 IdP에게 인증을 받은 후에는, 인증하는 엔티티(authenticating entity)는 전형적으로 쿠키, URL 다시쓰기(re-writing) 또는 다른 구현에 고유한 방식으로 사용자와의 세션을 설정할 것이다. IdP는 이 인증 사건을 기반으로, 이후 SP 또는 다른 신뢰하는 자들에게 주장들을 발급할 수 있다. 의지하는 자는 이 주장을 이용하여 사용자와 자신만의 세션을 설정할 수 있다.

이와 같은 상황에서, IdP는 세션 기관(session authority)으로 동작하고 의지하는 자들은 세션 참여자들(session participants)로 동작할 수 있다. 얼마간의 시간 후에, 사용자는 개별적인 세션 참여자와 또는 세션 기관에서 관리하는 지정된 세션에 속하는 모든 세션 참여자들과 자신의 세션을 종료시키기를 원할 수 있다. 전자의 경우는 이 표준 밖의 영역에 속한다. 그러나, 후자는 SAML 2.0 주장과 프로토콜 표준의 SAML Single Logout 프로토콜에서 이 프로파일을 사용함으로써 충족될 수 있다.

사용자 또는 사용자 세션을 종료시키려는 관리자는 세션 기관을 접촉함으로써 또는 개별적인 세션 참여자들을 접촉함으로써 이 “global” 세션을 종료시키는 것을 선택할 수 있다. 또한, 세션

기관으로 동작하는 어떤 IdP는 그것이 사용자에 대해 다른 IdP의 주장들에 대하여 의지하는 기관이 되는 상황에서는, 그 자신이 세션 참여자가 될 수 있다는 사실에 주의한다.

이 프로파일은 Single Logout 프로토콜이 SOAP 바인딩과 같이 동기적인(synchronous) 바인딩과 결합하는 것을 허용하여 또한 HTTP Redirect, POST 또는 Artifact 바인딩과 같이 비동기적인(asynchronous) "전-채널(front-channel)" 바인딩들과 결합되는 것을 허용한다. 전-채널 바인딩은, 예를 들어 사용자의 세션 상태가 쿠키의 형태로 단지 사용자 에이전트에만 존재하고 사용자 에이전트와, 세션 참여자 또는 세션 기관의 직접적인 상호작용이 필요한 경우에 요구될 수 있다. 아래에서 설명되는 것처럼, 세션 참여자는 가능하면, 이 프로파일을 기동시킬 때 "전-채널" 바인딩을 사용해야 한다. 이를 통해 세션 기관이 모든 참여자들에게 성공적으로 로그아웃을 전달할 수 있을 가능성을 최대화할 수 있다.

5.4.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:SSO:logout

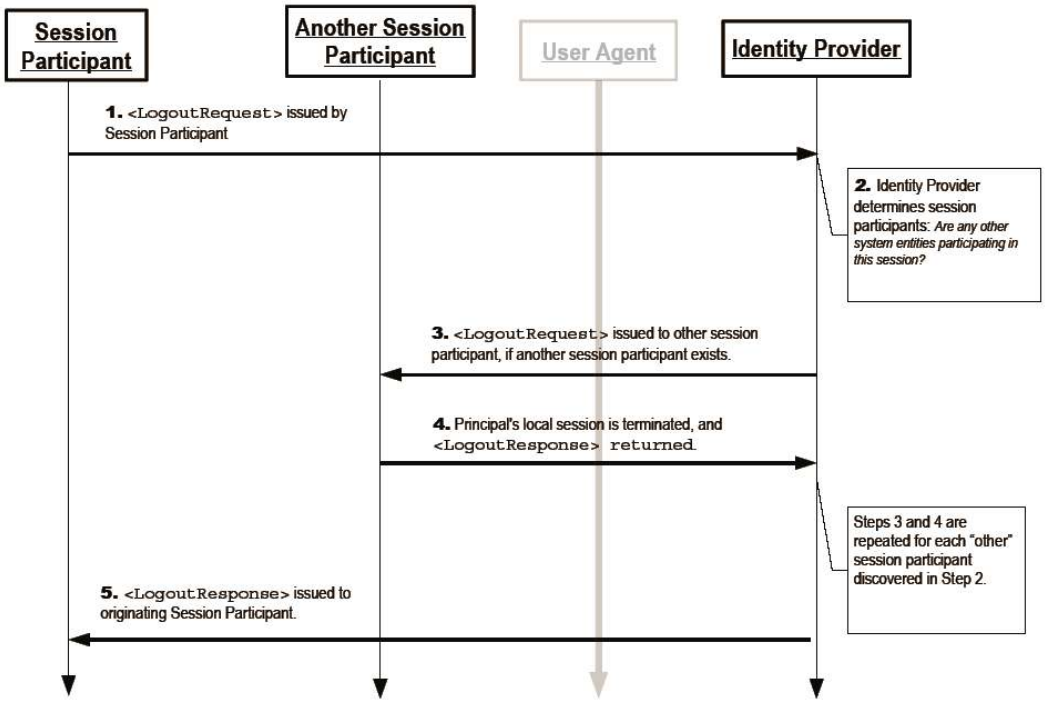
Contact information: security-services-comment@lists.oasis-open.org

Description: 아래에 주어짐.

Updates: 없음.

5.4.2. 프로파일 개요

(그림 5-3)은 단일 로그아웃(single logout)을 달성하기 위한 기본적인 템플릿을 도식화한 것이다.



(그림 5-3) 단일 로그아웃 프로파일 흐름도

사용자 에이전트를 회색으로 표시한 것은, 이 프로파일을 구현하는데 사용된 SAML 바인딩에 따라, 메시지 교환이 사용자 에이전트를 통해 이루어질 수도 있고 또는 시스템 엔티티들 간에 직접적으로 이루어질 수도 있음을 도식화 한 것이다.

다음 단계들은 이 프로파일에서 설명된다. 개별적인 단계 내에서, 그 단계에 대한 바인딩과 다른 구현 의존적인 행동들에 따라 하나 또는 그 이상의 실질적인 메시지 교환들이 있을 수 있다.

1. **세션 참여자가 IdP에게 발급하는 <LogoutRequest>**

   단계 1에서, 세션 참여자는 그것이 대응되는 인증 주장을 받은 IdP에게 <LogoutRequest> 메시지를 송신함으로써 단일 로그아웃을 기동시키고 사용자의 세션(들)을 종료시킨다. 요청은 IdP에게 직접 전달될 수도 있고 사용자 에이전트를 통해 간접적으로 전달될 수도 있다.

2. **IdP가 세션 참여자들을 결정**

   단계 2에서, IdP는 종료되고 있는 세션(들)을 결정하기 위해, <LogoutRequest> 메시지의 내용을 사용한다. 만약 IdP가 스스로 로그아웃을 기동시켰다면, 다른 메커니즘을 통해 종료되고 있는 세션(들)을 결정한다. 만약 다른 세션 참여자가 존재하지 않으면, 이 프로파일은 단계 5로 이동한다. 만약 그렇지 않다면, 단계 3과 단계 5가 식별되는 각각의 세션 참여자에 대하여 반복된다.

3. **IdP가 세션 참여자/기관에게 발급하는 <LogoutRequest>**

   단계 3에서, IdP는 종료되고 있는 하나 또는 그 이상의 세션(들)과 연관된 세션 참여자 또는 세션 기관에게 <LogoutRequest> 메시지를 발급한다. 이 요청은 단계 1의 요청 형태와 일관성이 있다면, 해당 엔티티에게 직접 전달되거나 또는 사용자 에이전트를 통해 간접적으로 전달될 수 있다.

4. **세션 참여자/기관이 IdP에게 <LogoutResponse> 발급**

   단계 4에서, 세션 참여자나 또는 세션 기관은 가능하면 요청이 지시한 대로 사용자의 세션(들)을 종료시키고 <LogoutResponse>를 IdP에게 반환한다. 응답은 단계 3의 요청 형태와 일관성이 있다면, IdP에게 직접 반환되거나 또는 사용자 에이전트를 통해 간접적으로 반환될 수 있다.

5. **IdP가 세션 참여자에게 <LogoutResponse> 발급**

   단계 5에서, IdP는 원래 로그아웃을 요청한 세션 참여자에게 <LogoutResponse> 메시지를 발급한다. 응답은 단계 1의 요청 형태와 일관성이 있다면, 세션 참여자에게 직접 반환되거나 또는 사용자 에이전트를 통해 간접적으로 반환될 수 있다.

세션 기관으로 동작하는 IdP는 단계 2에서 이 프로파일을 기동시킬 수 있고 모든 세션 참여자들에게 <LogoutRequest>를 발급할 수 있으며, 또한 단계 5를 건너(skip)뛸 수 있다.

5.4.3. 프로파일 설명

이 프로파일이 세션 참여자에 의해 시작된다면, 5.4.3.1절부터 시작한다. 만약 IdP에 의해

시작된다면, 5.4.3.2 절부터 시작한다. 아래 설명에서 다음이 참조된다.

Single Logout Service

이것은 IdP나 세션 참여자에 있는 단일 로그아웃 프로토콜 엔드포인트다. <LogoutRequest> 또는 <LogoutResponse> 메시지, 또는 그들을 나타내는 artifact가 이 프로토콜 엔드포인트로 배달된다. 요청과 응답에 대하여 동일한 엔드포인트가 사용될 수도 있고 다른 엔드포인트가 사용될 수 있다.

5.4.3.1. 세션 참여자가 IdP 에게 발급하는 <LogoutRequest>

만약 로그아웃 프로파일이 세션 참여자에 의해 시작된다면, 세션 참여자는 종료되는 세션(들)과 관련되어 자신이 수신했던 인증 주장(들)을 검사하고 IdP로부터 자신이 받았던 SessionIndex 값(들)을 수집한다. 만약 다중 IdP가 관련되어 있다면, 프로파일은 각각의 IdP에 대하여 독립적으로 반복되어야만 한다.

이 프로파일을 시작하기 위해, 세션 참여자는 하나 또는 그 이상의 적용 가능한 <SessionIndex> 요소들을 포함하는 <LogoutRequest> 메시지를 IdP의 단일 로그아웃 서비스 요청 엔드포인트에 발급한다. 적어도 하나의 요소가 포함되어야만 한다. 메타데이터가 이 엔드포인트의 위치와 IdP에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다.

비동기 바인딩(전-채널)

세션 참여자는, 만약 사용자의 사용자 에이전트가 존재한다면, HTTP Redirect, POST 또는 Artifact 바인딩과 같은 비동기 바인딩을 사용해서 사용자 에이전트를 통해 IdP에게 로그아웃 요청을 전달해야 한다. 그 다음에, IdP는 동기 또는 비동기 바인딩 중에 하나를 이용하여 다른 세션 참여자들에게 필요한 로그아웃 메시지를 전파시켜야 한다. 원래의 요청에 대하여 비동기 바인딩을 사용하는 것이 선호된다. 왜냐하면, 이것은 IdP가 단계 3동안에 다른 세션 참여자에게 성공적으로 로그아웃을 전파시킬 최고의 기회를 제공하기 때문이다.

만약 HTTP Redirect 또는 POST 바인딩이 사용된다면, <LogoutRequest> 메시지는 이 단계에서 IdP에게 배달된다. 만약 HTTP Artifact 바인딩이 사용된다면, 6장에서 정의된 Artifact Resolution 프로파일이 IdP에 의해 사용되며, 이것은 IdP가 예를 들어 SOAP 바인딩을 사용하여, <LogoutRequest> 메시지를 검색하기 위해 세션 참여자에게 콜백을 하게 한다.

이 단계에서 이루어지는 HTTP 교환들 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 이루어지도록 할 것이 권고된다. 만약 HTTP POST 또는 Redirect 바인딩이 사용된다면, <LogoutRequest> 메시지는 서명되어야만 한다. 만약 HTTP Artifact 바인딩이 사용된다면, 이것은 또한 artifact가 디레퍼런드될 때 요청 발급자를 인증하는 또 다른 수단을 제공한다.

이 바인딩들 각각은 세션 참여자가 원래의 요청과 프로파일 교환을 연관시킬 수 있도록 하는 RelayState 메커니즘을 제공한다. 세션 참여자는 프로파일의 사용이 프라이버시 조치들을 요구한다면, RelayState 값에 가능하면 최소한의 정보가 노출되도록 해야 한다.

동기 바인딩(후-채널)

다른 대안으로, 세션 참여자는 SOAP 바인딩과 같은 비동기적인 바인딩을 사용하여 요청을

IdP에게 직접 전송할 수 있다. 그 다음에, IdP는 동기 바인딩을 사용하여 다른 세션 참여자들에게 로그아웃 메시지를 전파해야 한다. 요청자는 <LogoutRequest>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 IdP에게 스스로를 인증해야만 한다.

<LogoutRequest> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.4.4.1절에 포함된다.

5.4.3.2. IdP 가 세션 참여자들을 결정

만약 로그아웃 프로파일이 IdP에 의해 시작되거나 또는 IdP가 유효한 <LogoutRequest> 메시지를 받았다면, IdP는 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼 요청을 처리한다. IdP는 식별자와 <SessionIndex> 요소를 검사하고 종료될 세션의 집합을 결정해야만 한다.

그 다음, IdP는 SAML 2.0 주장과 프로토콜 표준에서 설명하는 것처럼, 만약 존재한다면 최초의 요청 세션 참여자를 제외하고, 종료되는 세션에 참여하는 각각의 엔티티에 대하여 단계 3과 4를 수행한다.

5.4.3.3. IdP가 세션 참여자/기관에게 발급하는 <LogoutRequest>

로그아웃을 전파하기 위해, IdP는 자신의 <LogoutRequest>를 종료되는 세션에 속하는 세션 기관 또는 세션 참여자에게 발급한다. 이 요청은 응답자의 능력과 IdP에서 사용자 에이전트의 이용 가능성에 부합하는 SAML 바인딩을 사용하여 송신된다.

일반적으로, 단계 1에서 최초의 요청을 수신할 때 사용된 바인딩이 단계 1에서 기술한 내용을 제외하고는 이 단계에서 사용될 수 있는 바인딩을 결정하지는 않는다. 사용자 에이전트를 건너뛰는 동지 바인딩의 사용은 IdP가 추가적인 요청들을 전파시키기 위해 유사한 바인딩을 사용하도록 제약한다.

<LogoutRequest> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.4.4.1절에 포함된다.

5.4.3.4. 세션 참여자/기관이 IdP에게 <LogoutResponse> 발급

세션 참여자/기관은 SAML 2.0 주장과 프로토콜 표준에서 정의된 대로 <LogoutRequest> 메시지를 처리해야만 한다. 메시지를 처리한 후 또는 에러를 만나자마자, 이 엔티티는 SAML 프로토콜 교환을 완성하기 위해 요청하는 IdP에게 적절한 상태 코드를 포함하는 <LogoutResponse> 메시지를 발급해야만 한다.

동기 바인딩(후-채널)

만약 IdP가 SOAP 바인딩과 같은 동기 바인딩을 사용한다면, 응답은 동기 통신을 완료하기 위해 직접 IdP에게 반환된다. 응답자는 <LogoutResponse>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 이용하여, 자신을 요청하는 IdP에게 인증시켜야만 한다.

비동기 바인딩(전-채널)

만약 IdP가 HTTP Redirect, POST 또는 Artifact 바인딩과 같은 비동기 바인딩을 사용한다면, 그러면 <LogoutResponse> 또는 artifact가 사용자 에이전트를 통해 IdP의 단일 로그아웃 서비스 응답 엔드포인트로 반환된다. 메타데이터가 이 엔드포인트의 위치와 IdP에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다. 두 엔티티 모두에서 지원하는 어떠한 비동기 바인딩도 사용될 수 있다.

만약 HTTP Redirect 또는 POST 바인딩이 사용된다면, <LogoutResponse> 메시지는 이 단계에서 IdP에게 배달된다. 만약 HTTP Artifact 바인딩이 사용된다면, 6장에서 정의된 Artifact Resolution 프로파일이 IdP에 의해 사용되며, 이것은 IdP가 예를 들어 SOAP 바인딩을 사용하여, <LogoutResponse> 메시지를 검색하기 위해 세션 참여자에게 콜백을 하게 한다. 이 단계에서 이루어지는 HTTP 전송상에서 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 이루어지도록 할 것이 권고된다. 만약 HTTP POST 또는 Redirect 바인딩이 사용된다면, <LogoutResponse> 메시지는 서명되어야만 한다. 만약 HTTP Artifact 바인딩이 사용된다면, 이것은 또한 artifact가 디레퍼런드될 때 응답 발급자를 인증하는 또 다른 수단을 제공한다.

<LogoutResponse> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.4.4.2절에 포함된다.

5.4.3.5. IdP 가 세션 참여자에게 <LogoutResponse> 발급

이전 단계들에서 설명된 것처럼, 최초의 세션 발급자의 <LogoutRequest>을 처리한 후, IdP는 SAML 프로토콜 교환을 완료하기 위해 적절한 상태 코드를 포함하는 <LogoutResponse>를 가지고 최초의 요청에 대하여 응답해야만 한다.

이 응답은 최초의 요청에서 사용된 바인딩, 응답자의 능력(capability) 그리고 IdP에서의 사용자 에이전트의 이용 가능성과 부합하는 SAML 바인딩을 사용하여, 최초의 세션 참여자에게 전달된다. 단계 1에서 비동기 바인딩이 사용되었다면, 두 엔티티 모두가 지원하는 어떠한 바인딩도 사용될 수 있다.

<LogoutResponse> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.4.4.2절에 포함된다.

5.4.4. Signle Logout 프로토콜의 사용

이 절은 <LogoutRequest>와 <LogoutResponse> 사용법을 설명한다.

5.4.4.1. <LogoutRequest> 사용법

<Issuer> 요소는 반드시 존재해야 되며 요청하는 엔티티의 유일한 식별자를 포함해야만

한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다.

요청자는 응답자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

SAML 2.0 주장과 프로토콜 표준에서 정의된 매칭 규칙(matching rules)에 따라, 종료되는 세션과 관련하여 요청자가 발급했거나 또는 수신했던 인증 주장의 식별자와 강하게 일치하는 식별자를 사용하여 요청의 사용자가 반드시 식별되어야만 한다.

만약 요청자가 세션 참여자이면, 세션 참여자는 요청에 적어도 하나의 <SessionIndex> 요소를 포함해야만 한다. 만약 요청자가 세션 기관이거나 그것을 대신하여 동작하는 경우에는, 요청자는 모든 사용자의 가능한 세션들을 종료시키는 것을 가리키기 위해 이와 같은 요소들을 생략할 수 있다.

5.4.4.2. <LogoutResponse> 사용법

<Issuer> 요소는 반드시 존재해야 되며 응답하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다.

응답자는 요청자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

5.4.5. 메타데이터 사용

어떤 엔티티가 이 프로파일을 사용하여 요청과 응답을 전달할 수 있는 위치(들)를 설명하기 위해 하나의 엔드포인트 요소인 <md:SingleLogoutService>를 정의한다. 만약 사용자의 식별자를 암호화한다면, 요청자는 벌크 암호화 키를 전달하는데 사용하는 공개키와 함께, 사용하는 적절한 암호 알고리즘과 설정들을 결정하기 위해, encryption을 값으로 갖는 use 속성을 가지는 응답자의 <md:KeyDescriptor> 요소를 사용할 수 있다.

5.5. Name Identifier Management 프로파일

Name Identifier Management 프로파일에서 지원되는 시나리오에서, IdP는 SP와 사용자에 대한 영속적인(persistent) 형태의 식별자를 교환하며, 이를 통해 IdP와 SP가 어느 정도의 기간 동안 공통의 식별자를 소유할 수 있도록 해 준다. 그 후, IdP는 자신이 향후 동일한 사용자를 식별하는데 사용될 식별자의 형태와 값의 변화를 SP에게 알리기를 원할 수 있다. 또는, SP는 IdP가 향후 사용자에 대하여 SP와 통신할 때, IdP가 사용자에 대한 SP의 “alias”를 포함시킬 것을 보장하기 위해 사용자에 대한 자신의 “alias”를 첨가하기를 원할 수 있다. 마지막으로, IdP 또는 SP 중에 누구도 다른 제공자에게 자신이 더 이상 특정 식별자를 사용하는 메시지를

발급하거나 또는 수용하지 않을 것임을 알리기를 원할 수 있다. 이러한 시나리오들을 구현하기 위해, SAML Name Identifier Management 프로파일이 사용된다.

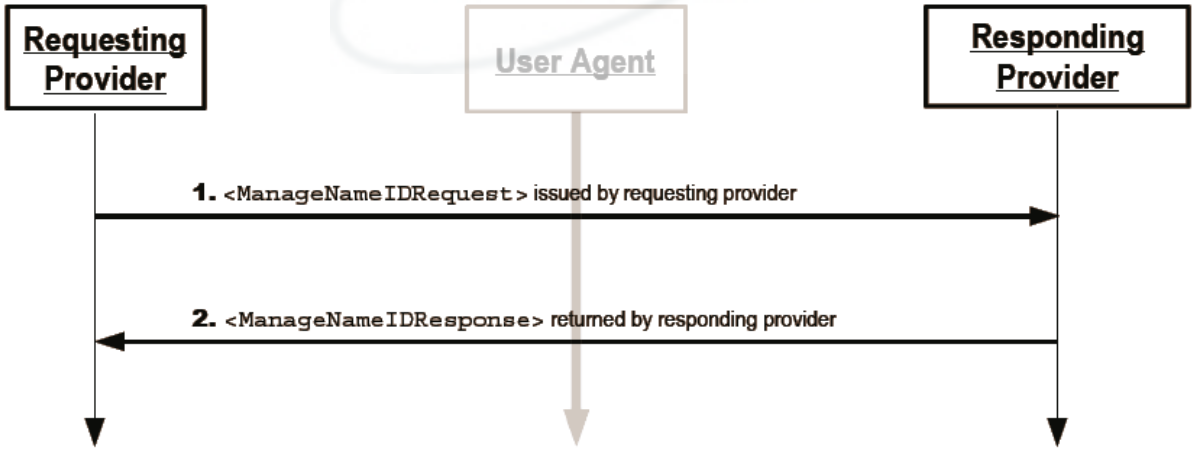
이 프로파일은 Name Identifier Management 프로토콜이 SOAP 바인딩과 같이 동기적인(synchronous) 바인딩과 결합하는 것을 허용하여 또한 HTTP Redirect, POST 또는 Artifact 바인딩과 같이 비동기적인(asynchronous) “전-채널” 바인딩들과 결합되는 것을 허용한다. 전-채널 바인딩은, 예를 들어, 사용자 에이전트와 응답하는 제공자간의 직접적인 상호작용이 변화를 달성하기 위해 필요한 경우에 요구될 수 있다.

5.5.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:SSO:nameid-mgmt
Contact information: security-services-comment@lists.oasis-open.org
Description: 아래에 주어짐.
Updates: 없음

5.5.2. 프로파일 개요

(그림 5-4)는 이름 식별자 관리(name identifier management) 프로파일에 대한 기본적인 템플릿을 도식화한 것이다.



(그림 5-4) 이름 식별자 관리 프로파일 흐름도

사용자 에이전트를 회색으로 표시한 것은, 이 프로파일을 구현하는데 사용된 SAML 바인딩에 따라, 메시지 교환이 사용자 에이전트를 통해 이루어질 수도 있고 또는 시스템 엔티티들 간에 직접적으로 이루어질 수도 있음을 도식화 한 것이다.

다음 단계들은 이 프로파일에서 설명된다. 개별적인 단계 내에서, 그 단계에 대한 바인딩과 다른 구현 의존적인 행동들에 따라 하나 또는 그 이상의 실질적인 메시지 교환들이 있을 수 있다.

1. 요청하는 IdP/SP가 <ManageNameIDRequest> 발급

단계 1에서, IdP나 또는 SP가 어떠한 변화를 알리기를 원하는 다른 제공자에게 <ManageNameIDRequest> 메시지를 전달함으로써, IdP나 또는 SP가 이 프로파일이 기동시킨다. 이 요청은 응답하는 제공자에게 직접 전달될 수도 있고 사용자 에이전트를 통해 간접적으로 전달될 수도 있다.

2. 응답하는 IdP/SP가 <ManageNameIDResponse> 발급

단계 2에서, 요청을 처리한 후, 응답하는 제공자는 최초의 요청하는 제공자에게 <ManageNameIDResponse> 메시지를 발급한다. 이 응답은 만약 단계 1의 요청 형태와 부합된다면, 요청하는 제공자에게 직접 전달될 수도 있고, 사용자 에이전트를 통해 간접적으로 전달될 수 도 있다.

5.5.3. 프로파일 설명

아래 설명에서 다음이 참조된다.

Name Identifier Management Service

이것은 IdP나 SP에 있는 이름 식별자 관리 프로토콜 엔드포인트다. <ManageNameIDRequest> 또는 <ManageNameIDResponse> 메시지, 또는 그들을 나타내는 artifact가 이 프로토콜 엔드포인트로 배달된다. 요청과 응답에 대하여 동일한 엔드포인트가 사용될 수도 있고 다른 엔드포인트가 사용될 수 있다.

5.5.3.1. 요청하는 IdP/SP 가 <ManageNameIDRequest> 발급

이 프로파일을 기동시키기 위해, 요청하는 제공자는 다른 제공자의 이름 식별자 관리 서비스 요청 엔드포인트에게 <ManageNameIDRequest>를 발급한다. 메타데이터가 이 엔드포인트의 위치와 응답하는 제공자에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다.

동기 바인딩(후-채널)

요청하는 제공자는 SOAP 바인딩과 같은 동기적인 바인딩을 사용하여 다른 제공자에게 요청을 직접 전달할 수 있다. 요청자는 <ManageNameIDRequest>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 다른 제공자에게 스스로를 인증해야만 한다.

비동기 바인딩(전-채널)

다른 대안으로, 요청하는 제공자는, 만약 사용자의 사용자 에이전트가 존재한다면, HTTP Redirect, POST 또는 Artifact 바인딩과 같은 비동기 바인딩을 사용해서 사용자 에이전트를 통해 다른 제공자에게 이 요청을 전달할 수 있다

만약 HTTP Redirect 또는 POST 바인딩이 사용된다면, <ManageNameIDRequest> 메시지는 이 단계에서 다른 제공자에게 배달된다. 만약 HTTP Artifact 바인딩이 사용된다면, 6장에서 정의된

Artifact Resolution 프로파일이 다른 제공자에 의해 사용되며, 이것은 다른 제공자가 예를 들어 SOAP 바인딩을 사용하여, <ManageNameIDRequest> 메시지를 검색하기 위해 요청 참여자에게 콜백을 하게 한다.

이 단계에서 이루어지는 HTTP 교환들이 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 이루어지도록 할 것이 권고된다. 만약 HTTP POST 또는 Redirect 바인딩이 사용된다면, <ManageNameIDRequest> 메시지는 서명되어야만 한다. 만약 HTTP Artifact 바인딩이 사용된다면, 이것은 또한 artifact가 디레퍼런드될 때 요청 발급자를 인증하는 또 다른 수단을 제공한다.

이 바인딩들 각각은 요청하는 제공자가 최초의 요청과 프로파일 교환을 연관시킬 수 있도록 하는 RelayState 메커니즘을 제공한다. 요청하는 제공자는 프로파일의 사용이 프라이버시 조치들을 요구한다면, RelayState 값에 가능하면 최소한의 정보가 노출되도록 해야 한다.

<ManageNameIDRequest> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.5.4.1절에 포함된다.

5.5.3.2. 응답하는 IdP/SP 가 <ManageNameIDRequest> 발급

수신자는 <ManageNameIDRequest> 메시지를 처리해야만 한다. 메시지 처리가 종료된 후 또는 에러를 만난 경우, 수신자는 SAML 프로토콜 교환을 완료하기 위해 요청하는 제공자에게 적절한 상태 코드를 포함하는 <ManageNameIDResponse> 메시지를 발급해야만 한다.

동기 바인딩(후-채널)

만약 요청하는 제공자가 SOAP 바인딩과 같은 동기적인 바인딩을 사용하였다면, 동기 통신을 완료하기 위해 하여 응답은 직접 반환된다. 응답자는 <ManageNameIDResponse>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 요청하는 제공자에게 스스로를 인증해야만 한다.

비동기 바인딩(전-채널)

만약 요청하는 제공자가 HTTP Redirect, POST 또는 Artifact 바인딩과 같은 비동기 바인딩을 사용하였다면, <ManageNameIDResponse> 또는 artifact가 사용자 에이전트를 통해 요청하는 제공자의 이름 식별자 관리 서비스 응답 엔드포인트로 반환된다. 메타데이터가 이 엔드포인트의 위치와 요청하는 제공자가 지원하는 바인딩들을 결정하는데 사용될 수 있다. 두 엔티티 모두가 지원하는 어떠한 바인딩도 사용될 수 있다.

만약 HTTP Redirect 또는 POST 바인딩이 사용된다면, <ManageNameIDResponse> 메시지는 이 단계에서 요청하는 제공자에게 배달된다. 만약 HTTP Artifact 바인딩이 사용된다면, 6장에서 정의된 Artifact Resolution 프로파일이 요청하는 제공자에 의해 사용되며, 이것은 다른 제공자가

예를 들어 SOAP 바인딩을 사용하여, <ManageNameIDResponse> 메시지를 검색하기 위해 요청하는 제공자에게 콜백을 하게 한다.

이 단계에서 이루어지는 HTTP 교환들이 기밀성과 메시지 무결성을 유지하기 위해 TLS 1.0 상에서 이루어지도록 할 것이 권고된다. 만약 HTTP POST 또는 Redirect 바인딩이 사용된다면, <ManageNameIDResponse> 메시지는 서명되어야만 한다. 만약 HTTP Artifact 바인딩이 사용된다면, 이것은 또한 artifact가 디레퍼런드될 때 응답 발급자를 인증하는 또 다른 수단을 제공한다.

<ManageNameIDResponse> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 5.5.4.2절에 포함된다.

5.5.4. Name Identifier Management 프로토콜의 사용

5.5.4.1. <ManageNameIDRequest> 사용법

<Issuer> 요소는 반드시 존재해야 되며 요청하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다.

요청자는 응답자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

5.5.4.2. <ManageNameIDReqesponse> 사용법

<Issuer> 요소는 반드시 존재해야 되며 응답하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다.

응답자는 요청자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

5.5.5. 메타데이터 사용

어떤 엔티티가 이 프로파일을 사용하여 요청과 응답을 전달할 수 있는 위치(들)를 설명하기 위해 하나의 엔드포인트 요소인 <md:ManageNameIDService>를 정의한다. 만약 사용자의 식별자를 암호화한다면, 요청자는 벌크 암호화 키를 전달하는데 사용하는 공개키와 함께, 사용하는 적절한 암호 알고리즘과 설정들을 결정하기 위해, encryption 을 값으로 갖는 use 속성을 가지는 응답자의 <md:KeyDescriptor> 요소를 사용할 수 있다.

6. Artifact Resolution 프로파일

SAML 2.0 주장과 프로토콜 표준은 SAML artifact 를 통해 대응되는 프로토콜 메시지를 디레퍼런싱하는 Artifact Resolution 프로토콜을 정의한다. HTTP Artifact 바인딩은 레퍼런스에 의해 SAML 프로토콜 메시지를 전달하기 위해 이 메커니즘을 활용한다. 이 프로파일은 SAML 2.0 바인딩 표준에서 정의된 SOAP 바인딩과 같은 동기 바인딩과 함께 이 프로토콜을 사용하는 것을 설명한다.

6.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:artifact

Contact information: security-services-comment@lists.oasis-open.org

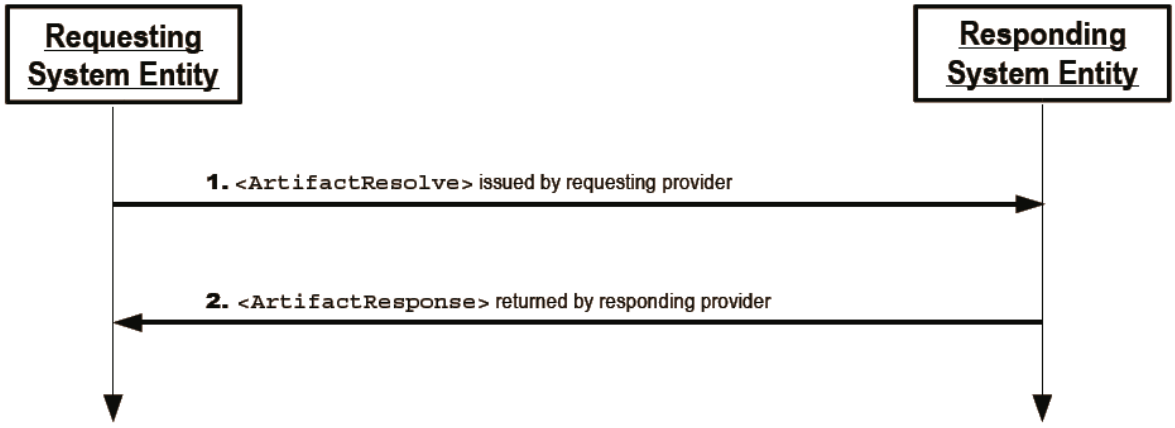
Description: 아래에 주어짐.

Updates: 없음

6.2. 프로파일 개요

이 프로파일을 지배하는 메시지 교환과 기본적인 처리 규칙은 교환되는 메시지들을 정의하는 SAML 2.0 주장과 프로토콜 표준에서 대부분 정의된다. SAML 2.0 바인딩 표준은 메시지 교환을 SOAP V1.1 로 바인딩하는 것을 정의한다. 이 부분에서 특별한 언급이 없다면, SAML 2.0 주장과 프로토콜 표준과 SAML 2.0 바인딩 표준에서 정의된 모든 요구사항들이 적용된다.

(그림 6-1)은 artifact 해결 프로파일에 대한 기본적인 템플릿을 도식화한 것이다.



(그림 6-1) artifact 해결 프로파일 흐름도

다음 단계들은 이 프로파일에서 설명된다.

1. 요청하는 엔티티가 <ArtifactResolve> 발급

단계 1에서, 요청자는 artifact 발급자에게 <ArtifactResolve> 메시지를 전송함으로써 이 프로파일을 기동시킨다.

2. 응답하는 엔티티가 <ArtifactResponse> 발급

단계 2에서, 요청을 처리한 후, 응답자는 <ArtifactResponse> 메시지를 요청자에게 발급한다.

6.3. 프로파일 설명

아래 설명에서 다음이 참조된다.

Artifact Resolution Service

이것은 <ArtifactResolve> 메시지가 배달되는 artifact 발급자에 있는 artifact 해결 프로토콜 엔드포인트다.

6.3.1. 요청하는 엔티티가 <ArtifactResolve> 발급

이 프로파일을 기동시키기 위해, 요청자가 artifact를 수신하고 SourceID를 사용하여 발급자를 결정한 후, 요청자는 artifact를 포함하는 <ArtifactResolve> 메시지를 artifact 발급자의 artifact 해결 서비스 엔드포인트로 전달한다. 메타데이터가 이 엔드포인트의 위치와 artifact 발급자에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다.

요청자는 SOAP 바인딩과 같은 동기적인 바인딩을 사용하여 artifact 발급자에게 요청을 직접 전달해야만 한다. 요청자는 <ArtifactResolve>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 응답자에게 스스로를 인증해야 한다. HPPT Artifact 바인딩을 사용하는 특정 프로파일은 인증을 의무화시키는 것과 같은 추가적인 요구사항을 가할 수 있다.

<ArtifactResolve> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 6.4.1절에 포함된다.

6.3.2. 응답하는 엔티티가 <ArtifactResponse> 발급

artifact 발급자는 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼 <ArtifactResolve> 메시지를 처리해야만 한다. 메시지를 처리한 후 또는 에러를 만나는 경우, artifact 발급자는 SAML 프로토콜 교환을 완료하기 위해 요청자에게 적절한 상태 코드를 포함하는 <ArtifactResponse> 메시지를 반환해야만 한다. 만약 처리가 성공이라면, artifact와 대응되는 디레퍼런스되는 SAML 프로토콜 메시지 또한 포함될 것이다.

응답자는 <ArtifactResponse>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 요청자에게 스스로를 인증해야만 한다.

<ArtifactResponse> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 6.4.2절에 포함된다.

6.4. Artifact Resolution 프로토콜의 사용

6.4.1. <ArtifactResolve> 사용법

<Issuer> 요소는 반드시 존재해야 되며 요청하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다.

요청자는 응답자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야 한다. HPPT Artifact 바인딩을 사용하는 특정 프로파일은 인증을 의무화시키는 것과 같은 추가적인 요구사항을 가할 수 있다.

6.4.2. <ArtifactResponse> 사용법

<Issuer> 요소는 반드시 존재해야 되며 응답하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 를 값으로 가져야만 한다.

응답자는 요청자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

6.5. 메타데이터 사용

SAML 2.0 메타데이터 표준은 지원되는 바인딩들과 요청자가 이 프로파일을 이용하여 요청을 전달할 수 있는 위치(들)를 설명하기 위해 하나의 엔드포인트 요소인 <md:ArtifactResolutionService>를 정의한다. index 속성이 artifact의 EndpointIndex 필드에서 참조에 의해 명기될 수 있는 가능한 엔드포인트들을 구별하기 위해 사용된다.

7. Assertion Query/Request 프로파일

SAML 2.0 주장과 프로토콜 표준은 참조나 또는 주체와 추가적인 문장에 고유한 기준에 따른 질의를 이용하여 이미 존재하는 주장들을 요청하기 위한 프로토콜을 정의한다. 이 프로파일은 SAML 2.0 바인딩 표준에서 정의된 SOAP 바인딩과 같은 동기 바인딩과 함께 이 프로토콜을 사용하는 것을 설명한다.

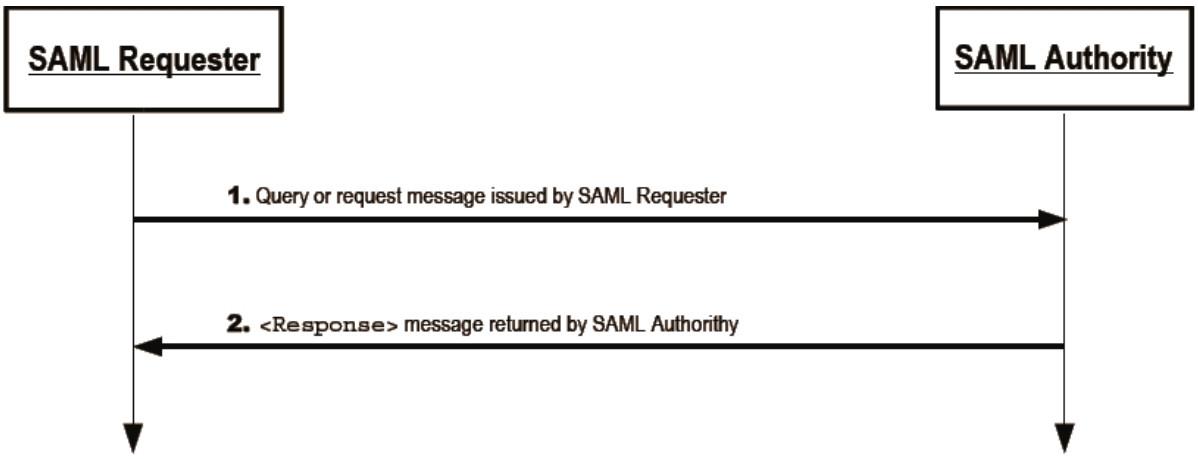
7.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:query
Contact information: security-services-comment@lists.oasis-open.org
Description: 아래에 주어짐.
Updates: 없음

7.2. 프로파일 개요

이 프로파일을 지배하는 메시지 교환과 기본적인 처리 규칙은 메시지를 교환하는데 사용되는 바인딩과 결합하여, 교환되는 메시지들을 정의하는 SAML 2.0 주장과 프로토콜 표준에서 대부분 정의된다. SAML 2.0 바인딩 표준은 메시지 교환을 SOAP V1.1로 바인딩하는 것을 정의한다. 이 부분에서 특별한 언급이 없다면, SAML 2.0 주장과 프로토콜 표준과 SAML 2.0 바인딩 표준에서 정의된 모든 요구사항들이 적용된다.

(그림 7-1)은 질의/요청 프로파일에 대한 기본적인 템플릿을 도식화한 것이다.



(그림 7-1) 질의/요청 프로파일 흐름도

다음 단계들은 이 프로파일에서 설명된다.

1. SAML 요청자가 Query/Request 발급

단계 1에서, SAML 요청자는 SAML 기관에게 <AssertionIDRequest>, <SubjectQuery>, <AuthnQuery>, <AttributeQuery> 또는 <AuthzDecisionQuery> 메시지를 전송함으로써 이 프로파일을 기동시킨다.

2. SAML 기관이 <Response> 발급

단계 2에서, 질의나 또는 요청을 처리한 후, 응답하는 SAML 기관은 <Response> 메시지를 SAML 요청자에게 발급한다.

7.3. 프로파일 설명

아래 설명에서 다음이 참조된다.

Query/Request Service

이것은 질의나 또는 <AssertionIDRequest> 메시지가 배달되는 SAML 기관에 있는 질의/요청 프로토콜 엔드포인트다.

7.3.1. SAML 요청자가 Query/Request 발급

이 프로파일을 기동시키기 위해, SAML 요청자는 <AssertionIDRequest>, <SubjectQuery>, <AuthnQuery>, <AttributeQuery> 또는 <AuthzDecisionQuery> 메시지를 SAML 기관의 질의/응답 서비스 엔드포인트에 발급한다. 메타데이터가 이 엔드포인트의 위치와 SAML 기관에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다.

SAML 요청자는 SOAP 바인딩과 같은 동기 바인딩을 사용하여 IdP에게 요청을 직접 전달해야만 한다. 요청자는 메시지를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 SAML 기관에게 인증해야 한다.

다양한 메시지들의 내용에 대한 프로파일에 고유한 규칙들은 7.4.1절에 포함된다.

7.3.2. SAML 기관이 <Response> 발급

SAML 기관은 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼 질의나 또는 요청 메시지를 처리해야만 한다. 메시지를 처리한 후 또는 에러를 만나는 경우, SAML 기관은 SAML 프로토콜 교환을 완료하기 위해 SAML 요청자에게 적절한 상태 코드를 포함하는 <Response> 메시지를 반환해야만 한다. 만약 요청이 하나 또는 그 이상의 일치하는 주장을 위치시키는데 성공한다면, 이들 주장들은 응답에 또한 포함될 것이다.

응답자는 <Response>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 요청자에게 스스로를 인증해야 한다.

<Response> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 7.4.2절에 포함된다.

7.4. Query/Request 프로토콜의 사용

7.4.1. Query/Request 사용법

<Issuer> 요소는 반드시 존재한다.

요청자는 응답자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야 한다.

7.4.2. <Response> 사용법

<Issuer> 요소는 반드시 존재해야 되며 응답하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 를 값으로 가져야만 한다. 이것이 반드시 반환되는 주장들의 <Issuer> 요소와 일치할 필요는 없다는 것에 주의한다.

응답자는 요청자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야 한다.

7.5. 메타데이터 사용

SAML 2.0 메타데이터 표준은 지원되는 바인딩들과 요청자가 이 프로파일을 이용하여 요청들이나 또는 질의들을 전달할 수 있는 위치(들)를 설명하기 위해 여러 개의 엔드포인트 요소들인 <md:AssertionIDRequestService>, <md:AuthnQueryService>, <md:AttributeService> 그리고 <md:AuthzService>를 정의한다.

만약 특정 엔티티에 대한 주장들이나 또는 주장 내용들을 암호화한다면, 벌크 암호화 키를 전달하는데 사용하는 공개키와 함께, 사용하는 적절한 암호 알고리즘과 설정들을 결정하기 위해, encryption 을 값으로 갖는 use 속성을 가지는 엔티티의 <md:KeyDescriptor> 요소를 사용할 수 있다.

다양한 역할 설명자들(role descriptors)이 특정 이름 식별자 포맷, 속성 프로파일들 또는 특정 속성들과 값들을 지원하는 일반적인 능력을 가리키기 위해 <md:NameIDFormat>, <md:AttributeProfile> 그리고 적용 가능한 <saml:Attribute> 요소들을 포함할 수 있다. 정해진 요청 동안에 이와 같은 특징을 지원하는 능력은 기관의 정책과 판단에 따라 결정된다.

8. Name Identifier Mapping 프로파일

SAML 2.0 주장과 프로토콜 표준은 하나의 사용자 이름 식별자를 동일한 사용자에 대한 다른 이름 식별자로 매핑하는 Name Identifier Mapping 프로토콜을 정의한다. 이 프로파일은 SAML 2.0 바인딩 표준에서 정의된 SOAP 바인딩과 같은 동기 바인딩과 함께 이 프로토콜을 사용하는 것을 설명한다. 이 프로파일은 암호화를 통해 사용자의 프라이버시를 보호하고 매핑된 식별자의 사용을 제약하는 것에 대한 추가적인 가이드라인을 설명한다.

8.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:nameidmapping

Contact information: security-services-comment@lists.oasis-open.org

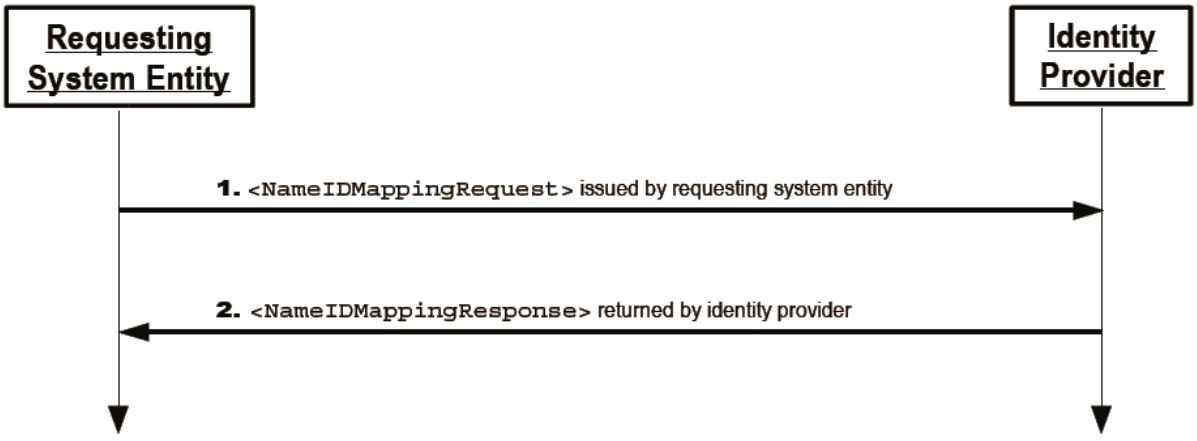
Description: 아래에 주어짐.

Updates: 없음

8.2. 프로파일 개요

이 프로파일을 지배하는 메시지 교환과 기본적인 처리 규칙은 메시지를 교환하는데 사용되는 바인딩과 결합하여, 교환되는 메시지들을 정의하는 SAML 2.0 주장과 프로토콜 표준에서 대부분 정의된다. SAML 2.0 바인딩 표준은 메시지 교환을 SOAP V1.1 로 바인딩하는 것을 정의한다. 이 부분에서 특별한 언급이 없다면, SAML 2.0 주장과 프로토콜 표준과 SAML 2.0 바인딩 표준에서 정의된 모든 요구사항들이 적용된다.

(그림 8-1)은 이름 식별자 매핑 프로파일에 대한 기본적인 템플릿을 도식화한 것이다.



(그림 8-1) 이름 식별자 매핑 프로파일 흐름도

다음 단계들은 이 프로파일에서 설명된다.

1. 요청하는 엔티티가 <NameIDMappingRequest> 발급

   단계 1에서, 요청자는 IdP에게 <NameIDMappingRequest> 메시지를 전송함으로써 이 프로파일을 기동시킨다.

2. IdP가 <NameIDMappingResponse> 발급

   단계 2에서, 요청을 처리한 후, 응답하는 IdP는 <NameIDMappingResponse> 메시지를 요청자에게 발급한다.

8.3. 프로파일 설명

이 절은 이름 식별자 매핑을 서비스를 사용한다. 이 서비스는 <NameIDMappingRequest> 메시지가 배달되는 IdP 에 있는 이름 식별자 매핑 프로토콜 엔드포인트다.

8.3.1. 요청하는 엔티티가 <NameIDMappingRequest> 발급

이 프로파일을 기동시키기 위해, 요청자는 <NameIDMappingRequest> 메시지를 IdP의 이름 식별자 매핑 서비스 엔드포인트에 발급한다. 메타데이터가 이 엔드포인트의 위치와 IdP에 의해 지원되는 바인딩들을 결정하는데 사용될 수 있다.

요청자는 SOAP 바인딩과 같은 동기 바인딩을 사용하여 IdP에게 요청을 직접 전달해야만 한다. 요청자는 <NameIDMappingRequest>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 IdP에게 인증해야만 한다.

<NameIDMappingRequest> 메시지들의 내용에 대한 프로파일에 고유한 규칙들은 8.4.1절에 포함된다.

8.3.2. IdP가 <NameIDMappingResponse> 발급

IdP는 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼 <ManageNameIDRequest> 메시지를 처리해야만 한다. 메시지를 처리한 후 또는 에러를 만나는 경우, IdP는 SAML 프로토콜 교환을 완료하기 위해 요청자에게 적절한 상태 코드를 포함하는 <NameIDMappingResponse> 메시지를 반환해야만 한다.

응답자는 <NameIDMappingResponse>를 서명하거나 또는 바인딩에서 지원하는 어떠한 다른 메커니즘을 사용하여 요청자에게 스스로를 인증해야만 한다.

<NameIDMappingResponse> 메시지의 내용에 대한 프로파일에 고유한 규칙들은 8.4.2절에 포함된다.

8.4. Name Identifier Mapping 프로토콜의 사용

SAML 2.0 주장과 프로토콜 표준은 한 사용자(principal)의 이름 식별자를 동일한 사용자에 대한 다른 이름 식별자로 매핑하는 이름 식별자 매핑 프로토콜을 정의한다. 이 절은 이 프로토콜의 사용과 매핑된 식별자의 사용을 제강하는 것과 같은 사용자의 프라이버시를 보호하기 위한 추가적인 가이드라인들을 설명한다.

8.4.1. <NameIDMappingRequest> 사용법

<Issuer> 요소는 반드시 존재한다.

요청자는 응답자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

8.4.2. <NameIDMappingResponse> 사용법

<Issuer> 요소는 반드시 존재해야 되며 응답하는 엔티티의 유일한 식별자를 포함해야만 한다; Format 속성은 반드시 생략되어야 하며, 만약 생략되지 않으면 urn:oasis:names:tc:SAML:2.0:nameid-format:entity를 값으로 가져야만 한다. 이것이 반드시 반환되는 주장들의 <Issuer> 요소와 일치할 필요는 없다는 것에 주의한다.

응답자는 요청자에게 자신을 인증시켜야 하며, 메시지를 서명하거나 또는 바인딩에 고유한 메커니즘을 사용하여 메시지 무결성을 보장해야만 한다.

SAML 2.0 주장과 프로토콜 표준에서 이름 식별자에 기밀성을 적용시키기 위해 암호를 사용하는 것을 정의하고 있다. 대부분의 경우, IdP는 사용자의 프라이버시를 보호하기 위해 그것이 요청자에게 반환하는 매핑된 이름 식별자를 암호화해야 한다. 요청자는 <EncryptedID> 요소를 추출하여 그 다음 프로토콜 메시지들 또는 주장들에 그것을 위치시킬 수 있다.

8.4.2.1. Mapped Identifier의 사용 제약

결과로 반환되는 식별자의 사용에 대한 추가적인 제약이 IdP에 의해 주장의 <Subject>에 식별자를 포함하지만 어떠한 문장도 없는 <Assertion> 형태로 매핑된 이름 식별자를 반환함으로써 적용될 수 있다. 그 다음, 이 주장은 암호화되고, 암호화된 결과가 요청자에게 반환되는 <EncryptedID>에 <EncyrptedData> 요소로 사용된다. 이 주장은 시간에 따른 제약들(time-based constraints) 또는 특정 의지하는 자들에 의한 사용 등과 같이 SAML 2.0 주장과 프로토콜 표준에서 정의된 것과 같이 사용을 제약하는 <Conditions> 요소를 포함할 수 있다. 이와 같은 주장은 무결성 보호를 위해 반드시 서명되어야만 한다.

8.5. 메타데이터 사용

SAML 2.0 메타데이터 표준은 지원되는 바인딩들과 요청자가 이 프로파일을 이용하여 요청들을 전달할 수 있는 위치(들)를 설명하기 위해 하나의 엔드포인트 요소인 <md:NameIDMappingService>를 정의한다.

만약 특정 엔티티에 대해 결과로 반환되는 식별자를 암호화한다면, 벌크 암호화 키를 전달하는데 사용하는 공개키와 함께, 사용하는 적절한 암호 알고리즘과 설정들을 결정하기 위해, encryption을 값으로 갖는 use 속성을 가지는 엔티티의 <md:KeyDescriptor> 요소를 사용할 수 있다.

9. SAML 속성 프로파일

속성 프로파일들은 특정 타입들의 속성 정보를 처리할 때나 또는 더 엄격한 처리를 요구하는 외부 시스템들과 상호작용을 할 때, SAML 속성 표현을 제약하는데 필요한 정의들을 제공한다. 이 절은 SAML 기본 속성 프로파일, X.500/LDAP 프로파일, UUID 프로파일들과 XACML 프로파일을 명기한다.

9.1. Basic 속성 프로파일

Basic 속성 프로파일은 내장된(built-in) XML 스키마 데이터 타입을 근거로 하는 속성 값들과 함께 단순화된 그러나 유일하지는 않은(non-unique) SAML 속성들의 명명(naming)을 명기하고, 이를 통해 문법을 검증하기 위해 확장 스키마가 필요한 경우를 제거한다.

9.1.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:attribute:basic
Contact information: security-services-comment@lists.oasis-open.org
Description: 아래에 주어짐.
Updates: 없음

9.1.2. SAML 속성 명명

<Attribute> 요소들에 있는 NameFormat XML 속성은 반드시 urn:oasis:names:tc:SAML:2.0:attrname-format:basic이어야만 한다.

Name XML 속성은 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼, 그 포맷에 대하여

명기된 규칙들을 따라야만 한다.

9.1.2.1. 속성 이름 비교

두 개의 <Attribute> 요소가 같은 SAML 속성을 참조하기 위한 필요충분 조건은 그들의 Name XML 속성들의 값들이 서로 같아야 한다.

9.1.3. 프로파일에 고유한 XML 속성들

<Attribute> 요소와 함께 사용될 목적으로 어떠한 XML 속성들도 정의되지 않는다.

9.1.4. SAML 속성 값들

<AttributeValue> 요소의 내용의 스키마 타입은 ITU-T X.1141 Annex A에서 정의된 타입들 중에 어떤 것으로부터 유도된 것이어야만 한다. xsi:type 속성은 존재해야 되며 적절한 값이 주어져야만 한다.

9.1.5. 예

```
<saml:Attribute
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic"
        Name="FirstName">
    <saml:AttributeValue xsi:type="xs:string">By-Tor</saml:AttributeValue>
</saml:Attribute>
```

9.2. X.500/LDAP 속성 프로파일

ITU-T X.500 표준들과 이 표준과 IETF RFC 3377에 기반을 둔 디렉터리들이 폭 넓게 채택되어 있다. 디렉터리 스키마는 이들 디렉터리들에 저장되어 있는 정보를 모델링하는데 사용된다. 특히, X.500에서는 속성 타입 정의들이 이 표준에서는 “디렉터리 속성들”이라고 언급하는 디렉터리에 있는 기본적인 정보 저장 단위인 속성들의 문법과 다른 특징들을 명기하는데 사용된다. 디렉터리 속성 타입은 X.500과 LDAP 표준들에 있는 스키마로 정의되고, inetOrgperson 스키마(IETF RFC 2798 참조)와 같은 같은 다른 공개 문서들의 스키마로 정의되고 그리고 사적인 목적들(private purposes)로 정의된 스키마로 정의된다. 이들 중 어떠한 경우에도, 배치자들이, 수작업으로 자신들을 위한 SAML에 고유한 속성 정의를 생성해야 할 필요 없이, SAML 속성 문들의 문맥으로 이들 디렉터리 속성 타입을 이용하고 그리고 이것이 상호운용

가능한 형태로 이루어지는 것은 유용한 것이다.

X.500/LDAP 속성 프로파일은 SAML 속성으로써 표현될 때, 이와 같은 속성들의 명명과 표현(representation)에 대한 공통적인 관례(convention)을 정의한다.

9.2.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:attribute:X509 (이것은 또한 대응되는 X.509/LDAP 프로파일 스키마 문서에 할당된 타깃 네임스페이스이다.)
Contact information: security-services-comment@lists.oasis-open.org
Description: 아래에 주어짐.
Updates: 없음

9.2.2. SAML 속성 명명

<Attribute> 요소들에 있는 NameFormat XML 속성은 반드시 urn:oasis:names:tc:SAML:2.0:attrname-format:uri 이어야만 한다.

속성 이름들을 구성하기 위해, IETF RFC 3061에서 설명된 URN oid 네임스페이스가 사용된다. 이 접근방식에서, Name XML 속성은 디렉터리 속성 타입에 할당된 OBJECT IDENTIFIER를 기반으로 한다.

예

```
urn:oid:2.5.4.3
```

X.500 절차들은 모든 속성 타입이 유일한 OBJECT IDENTIFIER로 식별되어야 하는 것을 요구하기 때문에, 이 명명 스키마는 유도된 SAML 속성 이름들이 애매하지 않다(unambiguous)는 것을 보장한다.

사람의 가동성을 위해, 일부 응용이 선택적인 문자열을 (IETF RFC 3061에서 정의된 것처럼) OID URN함께 이름을 전할 것을 또한 요구할 수 있다. SAML 2.0 주장과 프로토콜 표준에서 정의된 선택적인 XML 속성 FriendlyName은 이러한 목적으로 사용될 수 있다. 만약 디렉터리 속성 타입의 정의가 속성 타입에 대한 하나 또는 그 이상의 설명자들(짧은 이름들)을 포함한다면, FriendlyName 값은 만약 존재한다면, 정의된 설명자들 중에 하나이어야 한다.

9.2.2.1. 속성 이름 비교

두 개의 <Attribute> 요소가 같은 SAML 속성을 참조하기 위한 필요충분 조건은 그들의 Name XML 속성들의 값들이 IETF RFC 3061 에서 기술된 의미에서 서로 같아야 한다. FriendlyName 속성은 이 비교에서는 어떠한 역할도 수행하지 않는다.

9.2.3. 프로파일에 고유한 XML 속성들

<Attribute> 요소와 함께 사용될 목적으로 어떠한 XML 속성들도 정의되지 않는다.

9.2.4. SAML 속성 값들

원시적인(naive) X.500 디렉터리에서 사용될 목적으로 정의된 디렉터리 속성 타입 정의들은 ASN.1을 사용하여 속성의 문법을 명기한다. LDAP에서 사용할 목적으로, 디렉터리 속성 정의들은 추가적으로, LDAP에 고유한 인코딩으로 알려진, 어떻게 문법을 따르는 속성 또는 주장 값들이 LDAP 프로토콜로 전송될 때 표현될 수 있는지를 명기하는 LDAP 문법을 포함한다. LDAP에 고유한 인코딩은 일반적으로 UTF-8 형태로 Unicode 문자들을 생성한다. 이 SAML 속성 프로파일은 단지 LDAP 문법들을 가지는 이러한 디렉터리 속성들만을 위한 SAML 속성 값들의 형태를 명기한다. 이 프로파일에 대한 향후 확장에서는 디렉터리 속성의 문법들이 다른 인코딩을 명기하는 그러한 디렉터리 속성들에 대한 속성 값 형태들을 정의할 수도 있다.

특정 속성 값을 위해 사용되는 인코딩 규칙을 나타내기 위해, <AttributeValue> 요소는 반드시 XML 네임스페이스 urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500으로 정의된 Encoding으로 명명된 XML 속성을 포함해야만 한다.

그것의 LDAP에 고유한 인코딩이 배타적으로 UTF-8 문자 문자열들(character strings)을 값으로 생성하는 문법을 가진 어떠한 디렉터리 속성에 대해서도, SAML 속성 값은 어떠한 추가적인 공백(whitespace)를 가지고 있지 않는, <AttributeValue> 요소의 값으로서 단순히 UTF-8 문자열 자체로 인코딩된다. 이와 같은 경우에, xsi:type XML 속성은 **xs:string**으로 설정되어야만 한다. LDAP을 값으로 가지는 프로파일에 고유한 Encoding XML 속성이 제공된다.

이것이 적용되는 일부 LDAP 속성 문법(과 그것과 연관된 OID)의 리스트는 다음과 같다.

Attribute Type Description	1.3.6.1.4.1.1466.115.121.1.3
Bit String	1.3.6.1.4.1.1466.115.121.1.6
Boolean	1.3.6.1.4.1.1466.115.121.1.7
Country String	1.3.6.1.4.1.1466.115.121.1.11
DN	1.3.6.1.4.1.1466.115.121.1.12
Directory String	1.3.6.1.4.1.1466.115.121.1.15
Facsimile Telephone Number	1.3.6.1.4.1.1466.115.121.1.22
Generalized Time	1.3.6.1.4.1.1466.115.121.1.24
IA5 String	1.3.6.1.4.1.1466.115.121.1.26
INTEGER	1.3.6.1.4.1.1466.115.121.1.27
LDAP Syntax Description	1.3.6.1.4.1.1466.115.121.1.54
Matching Rule Description	1.3.6.1.4.1.1466.115.121.1.30

Matching Rule Use Description	1.3.6.1.4.1.1466.115.121.1.31
Name And Optional UID	1.3.6.1.4.1.1466.115.121.1.34
Name Form Description	1.3.6.1.4.1.1466.115.121.1.35
Numeric String	1.3.6.1.4.1.1466.115.121.1.36
Object Class Description	1.3.6.1.4.1.1466.115.121.1.37
Octet String	1.3.6.1.4.1.1466.115.121.1.40
OID	1.3.6.1.4.1.1466.115.121.1.38
Other Mailbox	1.3.6.1.4.1.1466.115.121.1.39
Postal Address	1.3.6.1.4.1.1466.115.121.1.41
Presentation Address	1.3.6.1.4.1.1466.115.121.1.43
Printable String	1.3.6.1.4.1.1466.115.121.1.44
Substring Assertion	1.3.6.1.4.1.1466.115.121.1.58
Telephone Number	1.3.6.1.4.1.1466.115.121.1.50
UTC Time	1.3.6.1.4.1.1466.115.121.1.53

다른 모든 LDAP 문법들에 대하여 속성 값은 <AttributeValue> 요소의 내용으로써, ASN.1 OCTET 문자열-인코드된 LDAP 속성 값을 포함하는(encompassing) base64-인코딩에 의해 인코드된다. xsi:type XML 속성은 xs:base64Binary로 설정되어야만 한다. "LDAP"을 값으로 가지는 프로파일에 고유한 Encoding XML 속성이 제공된다.

SAML 속성 값들에 대하여 동등성을 비교할 때, 대응되는 디렉터리 속성 타입에 대해 명기된 매칭 규칙들이 준수되어야만 한다. 이런 규칙에는 예를 들어 대소문자 구별(case sensitivity)가 있다.

9.2.5. 프로파일에 고유한 스키마

다음 스키마 리스팅은 프로파일에 고유한 Encoding XML 속성이 어떻게 타입은 ITU-T X.1141 Annex A에서 정의되어 있는지를 보인다.

```
<schema
        targetNamespace="urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500"
        xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="unqualified"
        attributeFormDefault="unqualified"
        blockDefault="substitution"
        version="2.0">
    <annotation>
        <documentation>
                Document identifier: saml-schema-x500-2.0
```

```
                Location: http://docs.oasis-open.org/security/saml/v2.0/
                Revision history:
                V2.0 (March, 2005):
                Custom schema for X.500 attribute profile, first published in
                SAML 2.0.
            </documentation>
        </annotation>
        <attribute name="Encoding" type="string"/>
</schema>
```

9.2.6. 예

다음은 “givenName” 디렉터리 속성의 매핑 예이다. 이 예에서는 SAML 주장 주체의 이름(first name)을 나타낸다. 이 속성의 OBJECT IDENTIFIER 는 2.5.4.42 이고 이것의 LDAP 문법은 Directory String 이다.

```
<saml:Attribute
        xmlns:x500="urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
        Name="urn:oid:2.5.4.42"
        FriendlyName="givenName">
    <saml:AttributeValue
            xsi:type="xs:string"
            x500:Encoding="LDAP">Steven</saml:AttributeValue>
</saml:Attribute>
```

9.3. UUID 속성 프로파일

UUID 속성 프로파일은 UUID 값들을 SAML 속성 이름들과 값들로 표현하는 것을 표준화한다. 속성의 소스 시스템(source system)dl 속성이나 속성의 값을 UUID를 가지고 식별하는 시스템일 때, 이 프로파일이 적용될 수 있다.

9.3.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:attribute:UUID

Contact information: security-services-comment@lists.oasis-open.org

Description: 아래에 주어짐.

Updates: 없음

9.3.2. SAML 속성 명명

<Attribute> 요소들에 있는 NameFormat XML 속성은 반드시 urn:oasis:names:tc:SAML:2.0:attrname-format:uri이어야만 한다.

만약 속성 이름을 표현하는 하부 표기법이 UUID이면, ITU-T Rec. X.667에서 설명된 URN uuid 네임스페이스가 사용된다. 이 접근 방식에서는, Name XML 속성은 속성을 식별하는 하부 UUID의 URN 형태를 기본으로 한다.

예

```
urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6
```

만약 속성 이름의 하부 표현이 UUID가 아니라면, 어떠한 형태의 URI도 Name XML 속성에서 사용될 수 있다.

사람의 가독성을 위해, 일부 응용이 URI와 함께 선택적인 문자열 이름을 전달하도록 또한 요구할 수 있다. 선택적인 XML 속성 FriendlyName이 이러한 목적으로 사용될 수 있다.

두 개의 <Attribute> 요소가 같은 SAML 속성을 참조하기 위한 필요충분 조건은 그들의 Name XML 속성 값들이 ITU-T Rec. X.667에서 기술된 의미에서 서로 같아야 한다. 속성 FriendlyName는 이 비료에서는 아무 역할도 수행하지 않는다.

9.3.3. 프로파일에 고유한 XML 속성들

<Attribute> 요소와 함께 사용될 목적으로 어떠한 XML 속성들도 정의되지 않는다.

9.3.4. SAML 속성 값들

속성의 값이 또한 UUID인 경우에, 위에서 설명한 것과 똑 같은 URN 문법이 <AttributeValue> 요소 내에서 값을 표현하는데 사용되어야만 한다. xsi:type XML 속성은 반드시 **xs:anyURI**로 설정되어야만 한다.

만약 속성의 값이 UUID가 아니라면, <AttributeValue> 요소의 사용에 대해서는 어떠한 제약도 있지 않다.

9.3.5. 예

다음은 "pre_auth_req" 설정인 DEC Extended Registry 속성의 예이다. "pre_auth_req" 설정은

6c9d0ec8-dd2d-11cc-abdd-080009353559를 값으로 갖는 잘 알려진 UUID 가지고 있으며 정수 값을 가진다.

```
<saml:Attribute
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
        Name="urn:uuid:6c9d0ec8-dd2d-11cc-abdd-080009353559"
        FriendlyName="pre_auth_req">
    <saml:AttributeValue xsi:type="xs:integer">1</saml:AttributeValue>
</saml:Attribute>
```

9.4. XACML 속성 프로파일

SAML 속성 주장들은 ITU-T Rec. X.1142에 따라 만들어지는 인가 결정에 대한 입력으로써 사용될 수 있다. SAML 속성 포맷이 XACML 속성 포맷과 다르기 때문에, 수행되어야만 하는 매핑이 존재한다. XACML 속성 프로파일은 명명과 값 문법 그리고 추가적인 속성 메타데이터를 표준화함으로써 이 매핑을 촉진시킨다. 이 프로파일에 따라 생성된 SAML 속성들은 자동적으로 XACML 속성들로 매핑될 수 있으며 XACML 인가 결정의 입력으로 사용될 수 있다.

9.4.1. 필요 정보

Identification: urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML(이것은 또한 대응되는 XACML 프로파일 스키마 문서에 할당된 타깃 네임스페이스이다.)
Contact information: security-services-comment@lists.oasis-open.org
Description: 아래에 주어짐.
Updates: 없음

9.4.2. SAML 속성 명명

<Attribute> 요소들에 있는 NameFormat XML 속성은 반드시 urn:oasis:names:tc:SAML:2.0:attrname-format:uri이어야만 한다.

Name XML 속성은 SAML 2.0 주장과 프로토콜 표준에서 정의된 것처럼, 그 포맷에 대하여 명기된 규칙들을 따라야만 한다.

사람의 가독성을 위해, 일부 응용이 OID URN과 함께 선택적인 문자열 이름을 전달하도록 또한 요구할 수 있다. SAML 2.0 주장과 프로토콜 표준에서 정의된 선택적인 XML 속성 FriendlyName이 이러한 목적으로 사용될 수 있다. 그러나 FriendlyName은 동등한 XACML 속성으로 변환될 수 없다.

두 개의 <Attribute> 요소가 같은 SAML 속성을 참조하기 위한 필요충분 조건은 그들의

Name XML 속성들의 값들이 이진 비교(binary comparison)에서 서로 같아야 한다. FriendlyName 속성은 이 비교에서는 어떠한 역할도 수행하지 않는다.

9.4.3. 프로파일에 고유한 XML 속성들

XACML은 각각의 속성이 명시적인 데이터 타입을 전달할 것을 요구한다. 이 데이터 타입 값을 제공하기 위해, DataType으로 불리는 URI 값을 갖는 새로운 XML 속성이 XML 네임스페이스 urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML에서 정의된다.

이 프로파일을 따라는 SAML <Attribute> 요소들은 네임스페이스로 한정된 DataType 속성을 포함해야만 한다. 만약 그렇지 않다면, 그 값은 http://www.w3.org/2001/XMLSchema#string으로 가정된다.

만약 비-표준 값들이 사용되면, 비-표준 DataType 값들을 가지고 매핑된 SAML 속성들을 소비할 각각의 XACML PDP 새로운 데이터 타입들을 지원하도록 확장되어야만 한다.

9.4.4. SAML 속성 값들

<AttributeValue> 요소 내용의 문법은 부모 <Attribute>에서 나타나는 프로파일에 고유한 DataType XML 속성으로 표현된 데이터 타입에 대응되어야만 한다. SAML 2.0 주장과 프로토콜 표준에서 정의된 타입들과 대응되는 데이터 타입들에 대하여, xsi:type XML 속성이 또한 <AttributeValue> 요소(들)에서 사용되어야 한다.

9.4.5. 프로파일에 고유한 스키마

다음 스키마 리스팅은 프로파일에 고유한 DataType XML 속성이 어떻게 타입은 ITU-T X.1141 Annex A에서 정의되어 있는지를 보인다.

```
<schema
        targetNamespace="urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML"
        xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="unqualified"
        attributeFormDefault="unqualified"
        blockDefault="substitution"
        version="2.0">
    <annotation>
        <documentation>
                Document identifier: saml-schema-xacml-2.0
```

```
                Location: http://docs.oasis-open.org/security/saml/v2.0/
                Revision history:
                V2.0 (March, 2005):
                Custom schema for XACML attribute profile, first published in
                SAML 2.0.
        </documentation>
    </annotation>
    <attribute name="DataType" type="anyURI"/>
</schema>
```

9.4.6. 예

다음은 “givenName” LDAP/X.500 속성의 매핑 예로써, SAML 주장 주체의 이름(first name)을 나타낸다. 이것은 또한, 다중 속성 프로파일들이 각기 호환될 수 있을 때, 단일 SAML 속성이 다중 속성 프로파일을 따를 수 있음을 설명한다.

```
<saml:Attribute
        xmlns:xacmlprof="urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML"
        xmlns:ldapprof="urn:oasis:names:tc:SAML:2.0:profiles:attribute:LDAP"
        xacmlprof:DataType="http://www.w3.org/2001/XMLSchema#string"
        ldapprof:Encoding="LDAP"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
        Name="urn:oid:2.5.4.42" FriendlyName="givenName">
    <saml:AttributeValue xsi:type="xs:string">By-Tor</saml:AttributeValue>
</saml:Attribute>
```

표준 작성 공헌자

표준 번호 : TTAS.IT-X1141_3

이 표준의 제정·개정 및 발간을 위해 아래와 같이 여러분들이 공헌하였습니다.

구분	성명	위원회 및 직위	연락처	소속사
과제 제안	조영섭	PG101 위원	042-860-6942 yscho@etri.re.kr	ETRI
표준 초안 제출	조영섭	PG101 위원	042-860-6942 yscho@etri.re.kr	ETRI
표준 초안 검토 및 작성	이석래	PG101 의장	02-405-5330 sllee@kisa.or.kr	KISA
	진승헌	PG101 부의장	042-860-1254 jinsh@etri.re.kr	ETRI
	백종현	PG101 간사	02-405-5423 jhbaek@kisa.or.kr	KISA
	조상래	연구원	042-860-6939 slcho@etri.re.kr	ETRI
		외 PG101 위원		
표준안 심의	정교일	공통기반기술위워회 의장	042-860-1920 kyoil@etri.re.kr	ETRI
	원유재	공통기반기술위원회 부의장	02-405-5360 yjwon@kisa.or.kr	KISA
	이필중	공통기반기술위원회 부의장	054-279-2232 pjl@postech.ac.kr	포항공대
	김응배	공통기반기술위원회 부의장	042-860-5296 ebkim@etri.re.kr	ETRI
		외 TC1 위원		
사무국 담당	김 선	팀 장	031-724-0080 skim@tta.or.kr	TTA
	오흥룡	과 장	031-724-0083 hroh@tta.or.kr	TTA

정보통신단체표준(국문표준)

SAML 2.0 프로파일

(Profiles for SAML 2.0)

발행인 : 한국정보통신기술협회

발행처 : 한국정보통신기술협회

463-824, 경기도 성남시 분당구 분당로 47

Tel : 031-724-0114, Fax : 031-724-0109

발행일 : 2006.12.