国 际 电 信 联 盟

ITU-T X.1141

国际电信联盟
电信标准化部门

(06/2006)

X系列：数据网、开放系统通信和安全性

电信安全

安全断言标记语言（SAML 2.0）

ITU-T X.1141建议书

国际电信联盟

ITU-T X系列建议书

数据网、开放系统通信和安全性

业务和设施	X.1-X.19
接口	X.20-X.49
传输、信令和交换	X.50-X.89
网络概貌	X.90-X.149
维护	X.150-X.179
管理安排	X.180-X.199
开放系统互连	
模型和记法	X.200-X.209
服务限定	X.210-X.219
连接式协议规范	X.220-X.229
无连接式协议规范	X.230-X.239
PICS书写形式	X.240-X.259
协议标识	X.260-X.269
安全协议	X.270-X.279
层管理对象	X.280-X.289
一致性测试	X.290-X.299
网间互通	
概述	X.300-X.349
卫星数据传输系统	X.350-X.369
以IP为基础的网络	X.370-X.379
报文处理系统	X.400-X.499
号码簿	X.500-X.599
OSI 组网和系统概貌	
组网	X.600-X.629
效率	X.630-X.639
业务质量	X.640-X.649
命名、寻址和登记	X.650-X.679
抽象句法记法1(ASN.1)	X.680-X.699
OSI 管理	
系统管理协议子集和结构	X.700-X.709
管理通信服务和协议	X.710-X.719
管理信息的结构	X.720-X.729
管理功能	X.730-X.799
安全	X.800-X.849
OSI 应用	
托付、并发和恢复	X.850-X.859
事务处理	X.860-X.879
远程操作	X.880-X.889
ASN.1的一般应用	X.890-X.899
开放分布式处理	X.900-X.999
电信安全	**X.1000-**

欲了解更详细信息，请查阅 *ITU-T*建议书目录。

ITU-T X.1141建议书

安全断言标记语言（SAML 2.0）

摘　要

SAML 是用于互换安全信息的基于 XML 的框架。该安全信息是以有关主体的断言的形式表示的，而主体是在一定的安全域中具有标识的实体（人或是计算机）。单一的断言可能包含有与认证、授权和属性相关的一些内部的不同说明。本建议书规定一种协议，通过该协议客户端可以从 SAML 授权中心请求断言并从它们那里获得响应。由基于 XML 的请求和响应消息格式组成的该协议可以承载在许多不同的下层通信和传输协议上；SAML 目前仅规定了一种在 HTTP 上承载 SOAP 的绑定。在生成它们的响应时，SAML 授权中心可以使用各种信息资源，如接收到的随请求输入的外部策略贮备和断言。本建议书规定 SAML 断言元素、主体、条件以及处理规则和说明。另外，本建议书规定了可充分理解的 SAML 元数据协议子集，这些协议子集包括相关的命名空间、通用的数据类型、处理规则和签字处理。同时规定了一些协议绑定，包括 SOAP、PAOS(反向 SOAP)、HTTP 重定向，HTTP POST 等。本建议书给出了可以使 SAML 2.0 在工业界被广泛采用的 SAML 的完全列表，包括 Web 浏览 SSO 协议子集和单次退出协议子集等。同时还给出了认证关联和一致性指南。

本建议书在技术上等同于并与 OASIS SAML 2.0 标准相兼容。

来　源

ITU-T 第 17 研究组（2005-2008）按照 ITU-T A.8 建议书规定的程序，于 2006 年 6 月 13 日批准了 ITU-T X.1141 建议书。

前　言

国际电信联盟（ITU）是从事电信领域工作的联合国专门机构。ITU-T（国际电信联盟电信标准化部门）是国际电信联盟的常设机构，负责研究技术、操作和资费问题，并且为在世界范围内实现电信标准化，发表有关上述研究项目的建议书。

每四年一届的世界电信标准化全会（WTSA）确定 ITU-T 各研究组的研究课题，再由各研究组制定有关这些课题的建议书。

WTSA 第 1 号决议规定了批准 ITU-T 建议书须遵循的程序。

属 ITU-T 研究范围的某些信息技术领域的必要标准，是与国际标准化组织（ISO）和国际电工技术委员会（IEC）合作制定的。

注

本建议书为简明扼要起见而使用的“主管部门”一词，既指电信主管部门，又指经认可的运营机构。

遵守本建议书的规定是以自愿为基础的，但建议书可能包含某些强制性条款（以确保例如互操作性或适用性等），只有满足所有强制性条款的规定，才能达到遵守建议书的目的。“应该”或“必须”等其他一些强制性用语及其否定形式被用于表达特定要求。使用此类用语不表示要求任何一方遵守本建议书。

知识产权

国际电联提请注意：本建议书的应用或实施可能涉及使用已申报的知识产权。国际电联对无论是其成员还是建议书制定程序之外的其他机构提出的有关已申报的知识产权的证据、有效性或适用性不表示意见。

至本建议书批准之日止，国际电联尚未收到实施本建议书可能需要的受专利保护的知识产权的通知。但需要提醒实施者注意的是，这可能并非最新信息，因此特大力提倡他们通过下列网址查询电信标准化局（TSB）的专利数据库：http://www.itu.int/ITU-T/ipr/。

目 录

ITU-T X.1141建议书

安全断言标记语言（SAML 2.0）

1 范围

本建议书规定安全断言标记语言（SAML 2.0）。SAML 规定由一个系统实体针对某一个主体所做的断言的句法和处理语义。在生成或使用这类断言的过程中，SAML 系统实体可以使用其他协议进行有关断言本身或断言主体的通信。本建议书规定 SAML 断言的结构、相关的协议集，另外也包含在管理 SAML 系统中的处理规则。

SAML 断言和协议消息使用 XML 编码并且使用 XML 命名空间。这些消息一般嵌入到其他的传输协议结构，如 HTTP POST 请求或 XML 编码的 SOAP 消息。本建议书也规定为嵌入和传输 SAML 协议消息提供框架的 SAML 绑定。此外，本建议书也提供为完成特定的使用场景或在使用 SAML 特征时实现互操作性而使用 SAML 断言和协议的协议子集的基线集。

本建议书规定如下内容：

1） SAML的一致性需求；

2） SAML的断言和协议：

- SAML断言Schema；
- SAML协议Schema。

3） SAML的绑定；

4） SAML协议子集：

- SAML ECP协议子集Schema；
- SAML X.500/LDAP属性协议子集Schema；
- SAML DCE PAC 属性协议子集Schema；
- SAML XACML 属性协议子集Schema；

5） SAML元数据；

6） SAML元数据Schema；

7） SAML认证关联。

2 参考文献

下列 ITU-T 建议书和其他参考文献的条款，在本建议书中的引用而构成本建议书的条款。在出版时，所指出的版本是有效的。所有的建议书和其他参考文献均会得到修订，本建议书的使用者应查证是否有可能使用下列建议书或其他参考文献的最新版本。国际电联电信标准化局保留当前有效的 ITU-T 建议书清单。IETF 保留 RFC 清单以及已经被后来的 RFC 替代的 RFC 的清单。W3C、Unicode 论坛和自由联盟保留最新的建议书及其他出版物的清单。

– ITU-T Recommendation X.660 (2004) | ISO/IEC 9834-1:2005, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: General procedures and top arcs of the ASN.1 Object Identifier tree.*

– ITU-T Recommendation X.667 (2004) | ISO/IEC 9834-8:2005, *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) | ISO/IEC 8824-1:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*
- ITU-T Recommendation X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications.*
- ITU-T Recommendation X.811 (1995) | ISO/IEC 10181-2:1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: authentication framework.*
- ITU-T Recommendation X.812 (1995) | ISO/IEC 10181-3:1996, *Information technology – Open Systems Interconnection – 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*.
- IETF RFC 1510 (1993), *The Kerberos Network Authentication Service (V5)*.
- IETF RFC 1750 (1994), *Randomness Recommendations for Security*.
- IETF RFC 1951 (1996), *DEFLATE Compressed Data Format Specification Version 1.3*.
- IETF RFC 1991 (1996), *PGP Message Exchange Formats*.
- IETF RFC 2045 (1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*.
- IETF RFC 2119 (1997), *Keywords for use in RFCs to Indicate Requirement Levels.*
- IETF RFC 2246 (1999), *The TLS Protocol Version 1.0*.
- IETF RFC 2253 (1997), *Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names*.
- IETF RFC 2396 (1998), *Uniform Resource Identifiers (URI): Generic Syntax*.
- IETF RFC 2535 (1999), *Domain Name System Security Extensions*.
- IETF RFC 2616 (1999), *Hypertext Transfer Protocol – HTTP/1.1*.
- IETF RFC 2617 (1999), *HTTP Authentication: Basic and Digest Access Authentication*.
- IETF RFC 2798 (2000), *Definition of the inetOrgPerson LDAP Object Class*.
- IETF RFC 2828 (2000), *Internet Security Glossary*.
- IETF RFC 2914 (2000), *Congestion Control Principles*.
- IETF RFC 2915 (2000), *The Naming Authority Pointer (NAPTR) DNS Resource Record*.
- IETF RFC 2945 (2000), *The SRP Authentication and Key Exchange System*.
- IETF RFC 2965 (2000), *HTTP State Management Mechanism*.
- IETF RFC 3023 (2001), *XML Media Types*.
- IETF RFC 3061 (2001), *A URN Namespace of Object Identifiers*.
- IETF RFC 3075 (2001), *XML-Signature Syntax and Processing*.
- IETF RFC 3377 (2002), *Lightweight Directory Access Protocol (v3): Technical Specification.*
- IETF RFC 3403 (2002), *Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database*.
- IETF RFC 3513 (2003), *Internet Protocol Version 6 (IPv6) Addressing Architecture*.
- IETF RFC 3546 (2003), *Transport Layer Security (TLS) Extensions*.
- IETF RFC 3923 (2004), *End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP)*.
- IETF RFC 4122 (2005), *A Universally Unique IDentifier (UUID) URN Namespace*.
- Liberty Alliance POAS:2003, R. Aarts, *Liberty Reverse HTTP Binding for SOAP Specification Version 1.0, Liberty Alliance Project*.
- OASIS WSS:2006, WS-Security Core Specification 1.1.
- UNICODE-C, M. Davis; M. J. Dürst: *Unicode Normalization Forms.* UNICODE Consortium, March 2001.
- W3C Canonicalization:2002, *Exclusive XML Canonicalization Version 1.0*, W3C Recommendation, Copyright © [18 July 2002] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut 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, Institut 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: Datatypes*, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut 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, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/.

注 — 本建议书引用某个文件，不意味着确认该文件自成一体时具备建议书的地位。

3 定义

就本建议书而言，下列定义适用。

3.1 引入定义

3.1.1 本建议书采用 ITU-T X.667 建议书规定的下列术语：

a） UUID。

3.1.2 本建议书采用 ITU-T X.680 建议书规定的下列术语：

a） 对象标识符；

b） 开放类型符号。

3.1.3 本建议书采用 ITU-T X.811 建议书规定的下列术语：

a） 责任人。

3.1.4 本建议书采用 ITU-T X.812 建议书规定的下列术语：

a） 接入控制消息；

b） 用户。

3.1.5 本建议书采用 W3C Web 业务词汇规定的下列术语：

a） 初始SOAP发送端；

b） 命名空间；

c） 最终SOAP接收端；

d） XML Schema。

3.1.6 本建议书采用 IETF RFC 2828 规定的下列术语：

a） 接入；

b） 接入控制；

c） 代理；

d） 代理服务器；

e） 拉；

f） 推；

g） 安全体系架构；

h） 安全策略；

i） 安全业务。

3.1.7 本建议书采用 IETF RFC 2396 规定的下列术语：

a） 统一资源标识符（URI）；

b） URI 参考。

3.2 附加定义

3.2.1 access rights 接入权利：一个主体可能具有的有关某一资源授权交互类型的描述。例子包括读、写、执行、增加、修改和删除。

3.2.2 account 账户：为在主要的和商务的服务提供商之间提供正规的交易和服务的一个正式的商务协议。

3.2.3 account linkage 账户链接：代表相同的责任人在两个不同的提供方处联系账户的方法，这样提供方可以针对该责任人进行交流。账户链接可以通过属性共享或通过标识联盟来建立。

3.2.4 active role 激活角色：系统实体在执行某些操作，例如接入某一资源时已经具有的一个角色。

3.2.5 administrative domain 管理域：由一个或多个管理策略、Internet 域名注册机构、国内合法实体（例如，个体、公司或其他正式的有组织的实体）的一个或多个组合以及主机、网络设备以及互连网络（以及可能的其他特征），加上（通常有各种）在其上运行的网络业务和应用。一个管理域可以包含或规定一个或多

个安全域。管理域可以包含一个单一的站点或多个站点。规定管理域的特征可能，并且在许多情况下，将会随着时间的进展而发展。管理域可能跨接管理域边缘通过提供和/或消费业务进行交互并达成一致。

3.2.6 administrator 管理者：搭建或维护系统的人或使用它来管理系统实体、用户和/或内容的人。管理者特别地与特定的管理域相关联并且可能与不止一个管理域相关联。

3.2.7 affiliation；affiliation group 从属关系，从属组：对于责任人的共享单一命名空间（以联合的形式）的系统实体集。

3.2.8 anonymity 匿名：匿名的量或状态，该量或状态是具有一个未知或隐藏的名字或标识的条件。

3.2.9 asserting party 断言方：正式地说，管理一个或多个 SAML 授权方的管理域。非正式地说，SAML 授权者的实例。

3.2.10 assertion 断言：由 SAML 授权者生成的与到针对一个主体执行的鉴权动作、有关于该主体的属性信息或应用于与特定资源相关的主体的授权数据相关的一些数据。

3.2.11 attribute 属性：对象的一个明确的特征。对于现实世界的主体，属性通常以物理特征规定，如大小、形状、重量和颜色。在安全空间的对象可能有描述大小、编码类型、网络地址等。属性通常以“属性名”和“属性值”对的形式表示，例如，“foo”有值“bar”，“count”有值“1”，“gizmo”有值“frob”和“2”。

3.2.12 attribute assertion 属性断言：承载与主体的属性相关的信息的断言。

3.2.13 attribute authority 属性授权中心：生成属性断言的系统实体。

3.2.14 authentication 认证：认证是指在一定程度的信任范围内，确定某人或某事事实上是其自身所宣称的人或事的过程。

3.2.15 authentication assertion 认证断言：承载有关针对某一主体的发生的成功认证结果的信息断言。

3.2.16 authentication authority 认证中心：生成认证断言的系统实体。

3.2.17 authorization 授权：通过评估可应用的接入控制信息来确定是否允许某主体具有接入到特定资源所规定的类型的过程。通常情况下，鉴权在认证关联文中进行。一旦某主体被认证，它可以被授权执行不同类型的接入。

3.2.18 authorization decision 授权决定：授权动作的结果。该结果可以是否定的，也就是说，它可以指示不允许该主体接入任何资源。

3.2.19 authorization decision assertion 授权决定断言：承载与认证决定相关的信息的断言。

3.2.20 back channel 反向信道：反向信道是指在两个系统实体之间不需要通过任何其他系统实体，如 HTTP 客户端（例如用户代理）进行“重定向”消息的直接通信。

3.2.21 binding，protocol binding 绑定，协议绑定：通常情况下，某些给定协议消息和消息互换模式到具体形式下的另一个协议的映射规范。例如，SAML `<AuthnRequest>`消息到 HTTP 的映射是绑定的一个例子。同一 SAML 消息到 SOAP 的映射是另一种绑定。在 SAML 关联中，每一种绑定以“SAML XXX 绑定”的模型给定一个名称。

3.2.22 credentials 信任状：为建立所宣称的主要标识传送的数据。

3.2.23 end user 端用户：为应用的目的利用资源的自然人。

3.2.24 entity 实体：见系统实体。

3.2.25 federate 联盟：将两个或多个实体联系或绑定在一起。

3.2.26 federation 联盟：在如下两种情况下使用该术语：

1） 在两个实体之间建立关系的动作；

2） 包含任何数量的服务提供商和标识提供方的关联。

3.2.27 federated identity 联合的身份：在用于指该主体的一系列标识和/或属性的提供方之间具有协议时，称一个责任人标识符在一系列提供者之间联合。

3.2.28 front channel 前向信道：前向信道指通过使用“HTTP 重定向”消息并通过用户代理，例如 Web 浏览器或任何其他 HTTP 客户端彼此之间传递消息，在两个使用 HTTP 进行交流的服务器之间形成的“通信信道”。

3.2.29 identifier 标识符：映射到专指该系统实体的一个系统实体的数据对象（例如，一个串）。一个系统实体可以具有表示该系统实体的多个不同的标识符。一个标识符本质上是一个实体的“特征属性”。

3.2.30 identity 身份：一个实体的本质。某一个身份通常使用其特征来描述，其中一些特征可以是任何数量的标识符。

3.2.31 identity defederation 解身份联盟：当提供方通过特定标识符和/或属性集同意停止参考一个责任人时发生的活动。

3.2.32 identity federation 身份联盟：代表一个责任人生成一个联合的身份的动作。

3.2.33 identity provider 身份提供者：为责任人生成、维护和管理身份信息并使用 Web 浏览器协议子集在一个联盟之中提供对其他服务提供商提供责任人认证的一种服务提供商。

3.2.34 identity provider lite 简易身份提供者：为责任人生成、维护和管理身份信息，并在一个联盟之中仅使用 SAML 所需要的部分提供责任人认证者。

3.2.35 login, logon, sign-on 登录：用户向一个认证中心提交信任状、建立简单会话并选择地建立丰富的会话的过程。

3.2.36 logout, logoff, sign-off 退出：一个用户表示希望中止一个简单的或丰富的会话的过程。

3.2.37 markup language 标记语言：为特定目的，应用于 XML 文件结构的一系列 XML 元素和 XML 属性。通过一系列 XML Schema 和相关的文件典型地规定标记语言。

3.2.38 name qualifier 名称限定词：消除一个可以用在多个命名空间（在联合的状态）来表示不同责任人没有歧义的一个串。

3.2.39 party 方：非正式地说，参与某一过程或通信（如接收一个断言或接入某一资源）的一个或多个责任人。

3.2.40 persistent pseudonym 持久的假名：为跨越多个会话的扩展时间区间由一个提供者分配的秘密保留名称标识符来向给定的信任方标识一个主体；持久的笔名可以用于表示一个身份联盟。

3.2.41 policy decision point (PDP) 策略决定点（PDP）：为其自身或为请求这样决定的其他系统实体进行认证决定的系统实体。例如，SAML PDP 吸收认证决定请求，并生成认证决定断言作为响应。PDP 是“策略决定中心”。

3.2.42 policy enforcement point (PEP) 策略执行点（PEP）：请求然后执行认证决定的系统实体。例如，SAML PEP 向 PDP 发送认证决定请求，然后吸收响应中发送的认证决定断言。

3.2.43 principal identity 责任人身份：责任人身份的一种表示，典型的是一个标识符。

3.2.44 profile 协议子集：用于几个目的之一的一系列规则；每一个系列以“SAML 的 xxx 协议子集”或“xxx SAML 协议子集”的模式给定一个名称。

1） 如何将断言嵌入到一个协议或使用的其他关联，并将其从一个协议或使用的其他关联中分离出来的规则。
2） 在使用的特定关联中使用SAML协议消息的规则。
3） 将以SAML表示的属性映射到另一个属性表示系统的规则。这样的一系列规则称为“属性协议子集”。

3.2.45 protocol binding 协议绑定：见“绑定”。

3.2.46 provider 提供者：表示身份提供者和服务提供商的一般方法。

3.2.47 relying party 信任方：根据来自另一个系统实体的信息决定采取动作的系统实体。例如，SAML 信任方依赖于从有关主体的断言方（SAML 权威）接收到的断言。

3.2.48 requester 请求端：利用 SAML 协议从另一个系统实体（SAML 授权中心，响应端）请求业务的系统实体。因为许多系统实体同时或串行地作为客户端和服务器，对于该概念不使用“客户端”术语表示。在 SAML 使用 SOAP 绑定的情况下，SAML 请求端在体系结构上区分于初始的 SOAP 发送端。

3.2.49 resource 资源：信息系统中包含的数据（例如，以文件的形式在存储器中的信息等），以及：

1） 系统提供的业务。

2） 一个系统设备（换句话讲，系统构件如硬件、固件、软件或文档）。

3.2.50 responder 响应端：一个系统实体（SAML 授权中心），该系统实体利用 SAML 协议响应来自另一个系统实体（请求端）对业务的请求。因为许多系统实体同时或串行地作为客户端和服务器。对于该概念不采用“服务器”作为术语名称。在将 SOAP 与 SAML 相绑定的情况，SAML 响应端在总体上与最终的 SOAP 接收端明显区分。

3.2.51 role 角色：字典中将角色定义为“表演者所扮演的部分或特征”或者“功能或位置”。系统实体连续地和/或同时扮演各种类型的角色，例如主动角色或被动角色。管理者的概念通常是一个角色的例子。

3.2.52 SAML artifact SAML 凭证：一个小的、固定长度的结构化数据主体，该数据主体指向典型的可变长 SAML 协议消息。SAML 凭证被设计成为嵌入到 URL 并在 HTTP 消息中传递，如带有“3xx 重定向”状态码的 HTTP 响应消息和后续的 HTTP GET 消息。以这种方式，服务提供商可以间接地通过用户代理将 SAML 凭证投递到另一个提供者，该提供者随后可以通过与代理提供者进行直接交互来废弃该 SAML 凭证并获得 SAML 协议消息。

3.2.53 SAML authority SAML 授权中心：SAML 域模型中的一个抽象系统实体，该系统实体签发断言。也见属性授权中心，认证授权中心和策略决定点（PDP）。

3.2.54 security 安全：确保信息的机密性的一系列安全措施，这些措施保护使用的系统或网络处理信息，同时控制接入它们。典型地，安全包括保密性、机密性、完整性和有效性。安全的目的是确保系统抵制相关的冲击。

3.2.55 security assertion 安全断言：在安全体系架构中细察的断言。

3.2.56 security context 安全关联：对于单个的 SAML 协议消息而言，消息的安全关联是消息安全头块（若有）连同在消息到接收端的投递中可能使用的其他安全机制的句法联合。对于后者，一个例子是在较低的网络栈层上使用的安全机制，如 HTTP、TLS 和 IPSec。

3.2.57 security domain 安全域：由安全模型和安全体系架构规定的环境或关联，包括接入到资源的鉴权的资源集和系统实体集。一个或多个安全域可以保留在单一的管理域中。随着时间的推移逐步解释给定的安全域特性。

3.2.58 security policy expression 安全策略表示：责任人身份和/或属性与其可允许的动作之间的映射。安全策略表示本质上通常是接入控制列表。

3.2.59 service provider 服务提供商：系统实体所代表的角色，在服务提供商处，系统实体向责任人或其他系统实体提供服务。

3.2.60 service provider lite 简易服务提供商：系统实体所代表的角色，在该角色处，系统实体仅使用 SAML 协议所需要的部分向责任人或其他系统实体提供业务。

3.2.61 session 会话：在系统实体之间持续的交互，通常包括责任人、为在持续交互期间维护某些交互状态所代表的系统实体。

3.2.62 session authority 会话授权者：在系统实体维护与会话相关的状态时所承担的角色。

3.2.63 session participant 会话参与方：当一个系统实体参与一个至少包含有一个会话授权者的会话时所承担的角色。

3.2.64 sign-off 退出：见“退出（logout）”。

3.2.65 sign-on 登录：见“登录（login）”。

3.2.66 site 站点：在地理上或 DNS 命名上用于一个管理域的非正式术语。它可以指一个管理域的特定地理或拓扑部分，或者它可以包含多个管理域，如像在 ASP 站点的情况。

3.2.67 subject 主体：在安全域的关联中的一个责任人。SAML 断言对主体进行相关的声明。

3.2.68 system entity; entity 系统实体，实体：计算机/网络系统的一个激活的元素。例如，一个自动的过程或过程集，一个子系统，或参与特定功能集的一个人或一组人。

3.2.69 time-out 超时：若某一事件没有发生，在某一条件变为真之后的一段时间。例如，由于某一会话在一段特定时间内处于非激活状态，使得其被终止，该会话被称谓“超时”。

3.2.70 transient pseudonym 临时笔名：由身份提供者指定的秘密保留的标识符用以向给定的信任方在不需要跨越多个会话的相对短的时间标识一个责任人。

3.2.71 XML attribute XML 属性：嵌入到 XML 元素的开始标记中并具有一个名字和值的一个 XML 数据结构。

3.2.72 XML element XML 元素：在一个 XML 文件中在其他这类结构中按体系安排的、并由一个开始标识和结束标识或者一个空标识指示的一个 XML 数据结构。

4 缩写

就本建议书而言，下列缩写适用。

AA	属性认证机构
ASN.1	抽象句法标记1
ASP	应用服务提供商
CA	认证机构
CMP	证书管理协议
CRL	证书撤销一览表
DCE	分布式计算环境
DDDS	动态授权发现系统
DNS	域名系统
ECP	增强的客户/代理
HTTP	超文本传送协议
HTTPS	安全超文本传送协议
IdP	身份提供者
IdP Lite	简易身份提供者
IP	网际协议
IPSEC	网际协议安全性
MD5	报文摘要算法5
MIME	多用途互联网邮件扩展
NAPTR	命名授权指示器
OID	对象标识符
PAC	特定属性证书
PAOS	反向SOAP
PDP	策略决定点
PEP	策略执行点
PGP	一种非对称加密软件
PKI	公开密钥基础设施
POP	拥有证据
RA	注册机构
RSA	Rivest、Shamir和Adleman公钥算法

SHA-1	安全散列算法1
SP	服务提供商
SPKI	简单公钥基础设施
SP Lite	简易服务提供商
SSO	单点登录
TLS	传输层安全协议
URI	通用资源标识符
UTC	协调的通用时间
UUID	通用唯一标识符
XACML	可扩展接入控制标识语言
XML	可扩展标识语言

5 惯例

本建议书使用关键字“务必”、“决不能”、“需要”、“须”、“不得”、“应”、“应不”、“建议”、“可”或“可选的”。在本建议书中，这些词应按 IETF RFC 2119 中所描述的解释。

本建议书使用符合 W3C XML Schema 第一部分，W3C XML Schema 第二部分以及描述 XML 编码 SAML 断言和协议报文的句法和语义的那些规范的标准文本的 XML Schema。在 SAML Schema 文本和本建议书中的 Schema 列表之间不一致时，以 Schema 文本为准。注意，在一些情况下，本建议书施加限制超出本 Schema 文本所指示的那些。

6 概述

本建议书规定安全断言标识语言版本 2(SAML v2.0)。本建议书规定由系统实体针对一个主体所做的断言的句法和处理语义。在生成了信任这样的断言的过程中，SAML 系统实体可以使用其他协议与相关的断言本身或断言的主体进行通信。本建议书规定 SAML 断言的结构以及相关的协议集加上在管理 SAML 系统中包含的处理规则。

SAML 断言和协议报文采用 XML 进行编码并使用 XML 命名空间。这些断言和协议报文典型地嵌入用于传输的其他结构中，如 HTTP POST 请求或 XML 编码的 SOAP 报文。第 7 节规定了 SAML 使用的公共数据类型。第 8 节提出了 SAML 断言协议的框架。第 9 节描述 SAML 元数据模型。第 10 节规定嵌入和传递 SAML 协议报文的框架。第 11 节提供使用 SAML 断言和协议实现特定使用场景或在使用 SAML 特征实现互操作性的协议子集的基线集。第 12 节讨论 SAML 的认证关联。特别地规定下列关联：

- SAML认证关联 Schema；
- SAML 认证关联 Schema类型；
- 用于网际协议的SAML关联类型 Schema；
- 用于网际协议密码的SAML关联类型 Schema；
- 用于Kerberos的SAML关联类型 Schema；
- 用于未注册的移动一因素的SAML关联类型 Schema；
- 用于未注册的移动二因素的SAML关联类型 Schema；
- 用于移动一因素合同的SAML关联类型 Schema；
- 用于移动二因素合同的SAML关联类型 Schema；
- 用于密码的SAML关联类别Schema；
- 用于密码保护传送的SAML关联类别Schema；
- 用于先前会话的SAML关联类别Schema；
- 用于公钥-X.509的SAML关联类别Schema；
- 用于公钥-PGP的SAML关联类别Schema；

- 用于公钥-SPKI的SAML关联类别Schema；
- 用于公钥-XML签字的SAML关联类别Schema；
- 用于智能卡的SAML关联类别Schema；
- 用于智能卡PKI的SAML关联类别Schema；
- 用于软件PKI的SAML关联类别Schema；
- 用于电话的SAML关联类别Schema；
- 用于电话的SAML关联类别Schema（游牧的）；
- 用于电话的SAML关联类别Schema（个性化的）；
- 用于电话的SAML关联类别Schema（认证的）；
- 用于安全远程密码的SAML关联类别Schema；
- 用于SL/TLS基于证书的客户端认证的SAML关联类别Schema；
- 用于时间同步令牌的SAML关联类别Schema。

第 13 节提供了为确保一致性需要遵循的 SAML 的框架。在第 13 节中讨论，包括操作方式和安全模式的一致性需求。附件 A 包括所有相关的 SAML Schema 列表。

7 通用数据类型

下面的节定义了如何使用和解释出现在 SAML（安全性断言标记语言） Schema 中的通用数据类型。

7.1 字符串值

所有的 SAML 字符串值都具有 **xs:string** 类型，它建立在 W3C XML Datatypes 中。除非在本建议书中有特别的说明，否则所有 SAML 消息中的字符串都必须包含至少一个非空的字符。

除非在本建议书中或者特定的协议子集中有明确的规定，否则，SAML 文件中具有 SAML Schema **xs:string** 类型或者从该类型衍生出来的类型的所有元素都必须通过一个确切的二进制对比方式进行对比。而且，SAML 执行以及部署不依赖对大小写不敏感的字符串对比方式、标准化或者修整空字段，或者本地特有的格式例如数量或者货币的转化。这个要求目的是要符合 W3C String 的规定。

如果一个执行情况是对比使用不同的字符编码来表示的值，执行情况必须采用一个可以返回和将两个值都转换为 Unicode 字符编码的方式相同的结果的对比方法，标准化格式 C，然后执行一个确定的二进制对比。这个要求目的是符合 W3C 字符模型，尤其是用于 Unicode 标准化文本的规则。

对比在 SAML 文件中接收的数据与来自外部资源的数据的应用必须考虑为 XML 规定的标准化规则。标准化包含在组件内的文本，因此通过换行符（ASCII CODE $10_{Decimal}$）来代表行的结束。定义为串（或者来自串的类型）的 XML 属性值按照 W3C XML1.0，第 3.3.3 节中的描述来进行标准化。利用 blank（ASCII CODE $32_{Decimal}$）来替代所有的空字符。

本建议书并没有为 XML 属性值或者元素内容定义整理或者排序顺序。SAML 执行情况一定不能依赖值的特定的排列顺序，因为根据涉及的主机的本地设置，它们有可能不同。

7.2 URI 值

所有的 SAML URI 参考值都具有类型 **xs:anyURI**，它建立在 W3C XML Datatypes 中。

除非在该建议书中有明确的说明，否则在 SAML 定义的元素或者属性中的所有 URI 参考值都必须包括至少一个非空字符，而且要求这些参考值必须完全。

本建议书扩展了 URI 参考作为标识符的使用情况，例如状态编码、格式类型、属性和相同实体名等。因此，对这些值的基本要求是唯一并且一致，也就是说同样的 URI 不能在不同时间代表不同的根本信息。

7.3 时间值

所有 SAML 时间值都具有 **xs:dateTime** 类型，它建立在 W3C XML Datatypes 中而且必须通过 UTC 格式来表达，它没有时区成分。

SAML 系统实体不应该指望时间解析精度高于毫秒。执行情况一定不能生成定义闰秒的时间值。

7.4 ID和ID参考值

xs:ID 简单的类型就是用来断言用于断言、请求和应答的 SAML 标识符。在本建议书中断言属于类型 **xs:ID** 的值除了必须满足 **xs:ID** 类型本身定义的属性外，还需要满足下面的属性：

- 分配一个标识符的任何一方必须确保那一方或任何其他方将偶然地把同样的标识符分配给不同的数据对象，具有一个不可忽略的概率。
- 一旦一个数据对象宣布它具有一个特定的标识符，那么必须有确定一个这样的声明。

SAML 系统实体保证标识符的唯一性的机制由具体的执行情况来定。此时采用了一种随机的或者伪随机的机制，两个随机选择的标识符相同的概率必须小于或者等于 2^{-128} 并且应该小于或者等于 2^{-160}。可以通过把一个随机选定的值编码在 128 到 160 比特长度之间来满足这个需求。编码方式必须符合 **xs:ID** datatype 中定义的规则。必须为一个伪随机码生成器输入唯一的内容，从而保证不同系统之间所需的唯一特性。

在 SAML 中利用 **xs:NCName** 简单类型来引用类型为 **xs:ID** 的标识符，因为 **xs:IDREF** 无法用于这个目的。在 SAML 中，通过 SAML 标识符引用指出的元素实际上可能没有在采用标识符的文档中进行定义，而是在另外的文档中有定义。采用 **xs:IDREF** 会妨碍它的值与同一个 XML 文档中某些元素中的 ID 属性值相匹配的需求。

8 SAML断言和协议

SAML 定义了由一个系统实体生成一个主体的断言的句法和处理语义。在生成或者信任这样一个断言的情况下，SAML 系统实体可以采用其他的协议来沟通断言本身或者一个断言的主体。本节定义了 SAML 断言的结构、相关的协议，同时还定义了管理一个 SAML 系统所涉及的处理规则。

SAML 断言和协议消息利用 XML 编码（参见 W3C XML1.0）并且采用 XML 的命名空间（参见 W3C Namespaces）。它们通常是嵌入到其他结构中进行传输例如 HTTP POST 请求或者 XML 编码的 SOAP 消息。第 10 节提供了 SAML 协议消息的嵌入和传输框架。第 11 节提供了使用 SAML 断言和协议的一个协议子集的基线集，从而在使用 SAML 特性的时候，实现特定使用情况或者实现的互通。

8.1 SAML断言

一个断言是一个信息包，该信息包提供了由 SAML 授权机构提出的 0 个或者多个断言；SAML 授权机构有时候是指讨论断言生成和交换的断言方以及通常被称做信赖方的利用接收到断言的系统实体。（这些术语和请求端和响应端不同，它们用于 SAML 协议信息交换的讨论。）

SAML 断言通常会针对一个主体而言，通过<subject>来表示。但是，<subject>元素可选，其他规范或者协议子集可能会采用 SAML 断言结构来生成元素似的断言而无需定义一个主体，或者通过另外一个可选的方式来定义一个主体。现在有一些服务提供商可以利用关于一个主体的断言来控制接入并且提供用户化的业务，因此他们会成为一个称做身份提供者的断言方的信赖方。

本建议书定义了由一个 SAML 授权机构生成的三个不同的断言。所有 SAML 定义的断言都和一个主体有关。在本建议书中定义的三个不同的断言如下：

- **认证**：断言主体在特定的时间由特定的方式来进行认证。
- **属性**：断言主体和提供的属性有关。
- **授权决定**：接收或者拒绝请求允许断言主体访问特定资源的请求。

注（资料性的）— PE13（参见OASIS PE:2006）建议在上一段中加上“或是不确定的”字样。

一个断言的外部结构是普通的，它提供的信息对它内部的所有断言来讲都一样。在断言的内部，一系列内部元素用来描述认证、属性、授权、决定或者用户定义的保护特定需求的断言。

SAML 断言 Schema 允许按第 8.6 节中的描述扩展，它允许用户对断言以及陈述定义扩展，同样也允许定义新的断言以及断言类别。

8.1.1 Schema头和命名空间断言

接下来的 Schema 段为断言 Schema 定义了 XML 命名空间以及其他头信息。

```
<schema targetNamespace="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="http://www.w3.org/2000/09/xmldsig#"
        schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-
20020212/xmldsig-core-schema.xsd"/>
    <import namespace="http://www.w3.org/2001/04/xmlenc#"
        schemaLocation="http://www.w3.org/TR/2002/REC-xmlenc-core-
20021210/xenc-schema.xsd"/>
    <annotation>
        <documentation>
            Document identity: saml-schema-assertion-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V1.0 (November, 2002):
              Initial Standard Schema.
            V1.1 (September, 2003):
              Updates within the same V1.0 namespace.
            V2.0 (March, 2005):
              New assertion schema for SAML V2.0 namespace.
        </documentation>
    </annotation>
…
</schema>
```

8.1.2 名字标识符

下面的内容定义了包含了主体的描述性标识符以及断言和协议消息发起者的 SAML 构造。

在 SAML 中有多种情况对两个系统实体来说与相关的第三方通信是有用的，例如，SAML 认证请求协议可以让第三方认证一个主体。因此，有必要建立一种方法，通过该机制每一方可以和对每个团体有意义的标识符关联起来。在某些情况下，需要限制一定的范围，在这个范围内标识符只对一个小的系统实体集有用（例如，保证一个主体的私密性）。同样的，标识符也可以用来指示一个 SAML 协议消息或者断言的发起者。

2 个或者多个系统实体可能会采用同样的名字标识符值来指示不同的实体。因此，每个实体对同样的名字可能会有不同的理解。SAML 通过把名字标识符放置在一个和名字限定词有关的联盟的命名空间里，来提供名字限定词从而消除一个名字标识符的歧义。SAML v2.0 根据一个断言方或者一个特定的信赖方或者联系来限定一个标识符，从而在需要的时候，允许标识符给出 pair-wise 语义。

还可以对名字标识符进行加密从而进一步提高它们的私密性特性，尤其是在可能通过中间媒介来对标识符进行传输的情况下。

注 — 为了避免使用相对高级的XML Schema结构，标识符元素的不同类型不会共享一个通用的类型层次。

8.1.2.1 <BaseID>元素

<BaseID>元素是一个扩展点，它允许应用增加新的标识符类型。它的 **BaseIDAbstractType** 复杂类型是摘要性质的，因此只能用在衍生出来的类型基础上。它包括下面的属性，扩展标识符表示法可以利用这些属性：

— NameQualifier [可选的]

限制标识符的安全或者管理域。这个属性提供了在没有冲突的情况下将不同用户存储的标识符联合起来的方法。

— SPNameQualifier [可选的]

通过一个服务提供商的名字或者提供者的联盟来进一步限制一个标识符。这个属性为在可信任方的基础上关联标识符提供了另外一个方法。

应该忽略 NameQualifier 和 SPNameQualifier 属性，除非标识符类型定义明确地定义了它们的用途和语义。

下面的 Schema 段定义了<BaseID>元素及其 **BaseIDAbstractType** 复杂类型：

```
<attributeGroup name="IDNameQualifiers">
    <attribute name="NameQualifier" type="string" use="optional"/>
    <attribute name="SPNameQualifier" type="string" use="optional"/>
</attributeGroup>
<element name="BaseID" type="saml:BaseIDAbstractType"/>
<complexType name="BaseIDAbstractType" abstract="true">
    <attributeGroup ref="saml:IDNameQualifiers"/>
</complexType>
```

8.1.2.2 复杂类型NameIDType

如果一个元素通过一个赋值的字符串名字来代表一个实体，采用 NameIDType 复杂类型。比起<BaseID>元素来讲，这是一个更加严格的标识符，它是在<NameID> 和<Issuer>元素下面的一个类型。除了包含实际标识符的字符串内容，它还提供下面可选的属性：

— NameQualifier [可选的]

限制名字的安全或者管理域。这个属性提供了一个把不同用户存储的名字，在避免冲突的情况下联合到一起的方法。

— SPNameQualifier [可选的]

利用服务提供商的名字或者提供者联盟来进一步限制一个名字。这个属性提供了在可信任方的基础上关联名字的一个额外的方法。

— Format [可选的]

代表基于字符串标识符信息的分类的 URI 参考。对于可以用做 Format 属性值以及它们相关描述和处理规则值的 SAML 定义的 URI 引用的详细信息参见第 8.7.3 节。除非由一个元素根据这个类型而进行了定义，否则如果没有提供 Format 值，那么值 urn:oasis:names:tc:SAML:1.0:nameid-format:unspecified（参见第 8.7.3.1 节）开始起作用。

如果采用了第 8.7.3 节中定义的一个格式值而不是其他的，那么可以根据本建议书之外提供的对格式的定义来对这个类型的元素的内容进行解释。否则如果不是由格式的定义来指示，那么匿名，使用假名的发布以及坚持使用和断言以及可信任方有关的标识符都是和特定执行情况有关。

— SPProvidedID [可选的]

若它和元素的内容中给出的原来名字标识符不同，服务提供商或者提供者联盟为一个实体建立的一个名字标识符。这个属性提供了一个把 SAML 以及现存的由服务提供商已经在使用的标识符集成起来的方法。例如，可以通过在第 8.2.8 节中定义的名字标识符管理协议把一个现存的标识符“附加”到实体上。

利用这个类型的元素，通过特定的 Fromat 定义为这些属性的内容定义附加的规则。如果元素或者格式没有明确地定义 NameQualifier 和 SPNameQualifier 属性的使用以及语义，那么忽略它们。

下面的 Schema 段定义了 **NameIDType** 复杂类型：

```
<complexType name="NameIDType">
    <simpleContent>
        <extension base="string">
            <attributeGroup ref="saml:IDNameQualifiers"/>
            <attribute name="Format" type="anyURI" use="optional"/>
            <attribute name="SPProvidedID" type="string" use="optional"/>
        </extension>
    </simpleContent>
</complexType>
```

8.1.2.3 <NameID>元素

<NameID>元素是类型 **NameIDType**（参见第 8.1.2.2 节），可以用在不同的 SAML 断言结构中，例如<Subject>和<SubjectConfirmation>元素以及不同的协议消息中使用（参见第 8.2 节）。

下面的 Schema 段定义了<NameID>元素：

```
<element name="NameID" type="saml:NameIDType"/>
```

8.1.2.4 <EncryptedID>元素

<EncryptedID>元素属于 **EncryptedElementType** 类型，它通过加密的方式携带未加密的标识符元素内容，参见 W3C 加密的定义。<EncryptedID>元素包括下面的元素：

— <xenc:EncryptedData>[必需的]

加密的内容以及相关的机密细节参见 W3C 加密中的定义。需要出现类型属性并且如果出现，必须包含一个 http://www.w3.org/2001/04/xmlenc#Element。加密的内容必须包含一个元素，该元素具有一个 **NameIDType** 或者 **AssertionType** 类型，或者从 **BaseIDAbstractType, NameIDType** 或者 **AssertionType** 衍生出来的一个类型。

— <xenc:EncryptedKey> [0 或者多个]

包装好的解密密钥，参见 W3C 加密中的定义。每个包装好的密钥应该包括一个 Recipient 属性，该属性指定为之加密密钥的实体。接收属性的值应该是一个 SAML 系统实体的 URI 标识符，参见第 8.4 节中的定义。

加密标识符的目的是当纯文本值穿越一个中间媒介的时候，它可以作为一个私密性保护机制。因此，对于任何加密操作来讲，密码应该都是唯一的。想要了解更多这方面的内容，参见 W3C XML 加密，第 6.3 节中的描述。

完整的断言可以加密到这个元素中并且用做一个标识符。在这种情况下，加密断言的<Subject>元素提供封装断言的主体的“标识符”。因此，如果标识断言无效，那么整个封装的断言均无效。

接下来的 Schema 段定义了<EncryptedID>元素及其 **EncryptedElementType** 复杂类型：

```
<complexType name="EncryptedElementType">
    <sequence>
        <element ref="xenc:EncryptedData"/>
        <element ref="xenc:EncryptedKey" minOccurs="0"
maxOccurs="unbounded"/>
    </sequence>
</complexType>
<element name="EncryptedID" type="saml:EncryptedElementType"/>
```

8.1.2.5 <Issuer>元素

<Issuer>元素和复杂类型 **NameIDType** 一起提供关于 SAML 断言或者协议消息的发起者的信息。元素要求利用一个字符串来携带发起者的名字，但是允许携带不同的描述性的数据（参见第 8.1.2.2 节）。

除了这个元素类型的基本规则，如果没有为这个元素提供 Format 值，那么值 urn:oasis:names:tc:SAML:2.0:nameid-format:entity 起作用（参见第 8.1.2.2 节）。

接下来的 Schema 段定义了<Issuer>元素：

```
<element name="Issuer" type="saml:NameIDType"/>
```

8.1.3 断言

接下来的各节定义了包含断言信息或者提供指示一个现存断言方式的 SAML 结构。

8.1.3.1 <AssertionIDRef>元素

<AssertionIDRef>元素通过它唯一的标识符来引用一个 SAML 断言，但是引用内容没有规定发起断言或者可以得到断言的特定机构。一个协议元素利用这样一个引用来请求相应的断言的具体情况，参见第 8.2.3 节。

接下来的 Schema 段定义了<AssertionIDRef>元素：

```
<element name="AssertionIDRef" type="NCName"/>
```

8.1.3.2 <AssertionURIRef>元素

<AssertionURIRef>元素通过 URI 引用来引用一个 SAML 断言。通过 URI 引用的一个特定方式，利用它可以重新获得相应的断言。在一个协议中，这个元素如何实现上述任务，参见第 7.3 节。

下面的 Schema 段定义了<AssertionURIRef>元素：

```
<element name="AssertionURIRef" type="anyURI"/>
```

8.1.3.3 <Assertion>元素

<Assertion>元素是 **AssertionType** 复杂类型。本类型定义了对所有断言来讲通用的信息，包括下面的元素和属性：

— <Version> [必需的]

改断言的版本。本建议书中定义了标识符中的 SAML 的版本是“2.0”。SAML 的版本问题参见第 8.3 节中的讨论。

— ID [必需的]

本断言的标识符。它的类型是 **xs:ID**，并且为了保持标识符的唯一性，必须服从第 7.3 节中定义的需求。

— IssueInstant [必需的]

UTC 中问题出现的时间点，参见第 7.3 节中的描述。

— <Issuer> [必需的]

断言中发起需求的 SAML 授权机构。对于目的可信任方来讲，发(若有）之间的特定的关系。任何由需要断言的信赖方或者通过特定的协议子集提出的需求都是和应用有关的。

— <ds:Signature> [可选的]

一个 XML 签字用来保护断言发起者的完整性并且对它进行验证，参见下面以及第 8.4 节中的描述：

— <Subject> [可选的]

断言中声明的主体。

— <Conditions> [可选的]

在使用断言的时候或者评估断言的有效性的时候，必须对当时的条件进行评估。关于如何评估条件的附加信息，参见第 8.1.5 节。

— <Advice> [可选的]

与断言有关的额外的信息，它有助于特定环境下的处理过程，但是如果应用无法理解建议或者说应用不希望采用该建议，那么应用就可以忽略它。

存在 0 个或者多个声明元素：

— <Statement>

在扩展 Schema 中定义的一个断言类型。必须利用 **xsi:type** 属性来指示实际的声明类型。

— <AuthnStatement>

认证说明。

— <AuthzDecisionStatement>

授权决定断言。

— <AttributeStatement>

属性断言。

没有断言元素的断言必须包含一个<subject>元素。这样一个断言定义了一个责任人，可以利用 SAML 方式来引用或者证实该责任人，但是断言没有更多的信息和该责任人相关联。

否则，如果存在<Subject>，它就用来标识断言中所有断言的主体。如果省略了<Subject>，那么断言中的断言应用到通过应用或者协议子集所特有的方式来标识主体上。SAML 本身没有定义这样的断言，而且一个没有主体的断言在本建议书中不存在一个已定义的含义。

根据特定协议或者协议子集的需求，经常需要对一个 SAML 断言的发起者进行认证，并且经常会需要完整性保护。在传递一个断言的时候，可以通过映射的协议所提供的机制来提供认证和对消息完整性的保护（参见第 10 节）。可以对 SAML 断言进行签字，它提供对发起者的保护和对消息的完整性保护。

如果采用了这样一个签字，那么必须存在<ds:Signature>元素，而且一个可信任方必须根据 W3C XML Signature 的规定来验证这个签字的有效性（也就是说该断言没有被篡改）。如果该签字无效，那么可信任方不能依赖断言的内容。如果签字有效，那么可信任方就应该对签字进行评估以确定发起者的身份和适合性，可以根据该建议书的规定继续处理断言并且认为它是适合的（例如，评估条件、建议、是否遵守协议子集特定的规则等）。

不管是否签字，包含多个断言的一个断言和包含单个断言的多个断言（分别提供主体、条件等）在语义上是等效的。

下面的 Schema 段定义了<Assertion>元素及其 **AssertionType** 复杂类型：

```
<element name="Assertion" type="saml:AssertionType"/>
<complexType name="AssertionType">
       <sequence>
              <element ref="saml:Issuer"/>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="saml:Subject" minOccurs="0"/>
              <element ref="saml:Conditions" minOccurs="0"/>
              <element ref="saml:Advice" minOccurs="0"/>
              <choice minOccurs="0" maxOccurs="unbounded">
                     <element ref="saml:Statement"/>
                     <element ref="saml:AuthnStatement"/>
                     <element ref="saml:AuthzDecisionStatement"/>
                     <element ref="saml:AttributeStatement"/>
              </choice>
       </sequence>
       <attribute name="Version" type="string" use="required"/>
       <attribute name="ID" type="ID" use="required"/>
       <attribute name="IssueInstant" type="dateTime" use="required"/>
</complexType>
```

8.1.3.4 <EncryptedAssertion>元素

<EncryptedAssertion>元素利用加密的方式表示一个断言，参见 W3C Encryption 中的定义。

<EncryptedAssertion>元素包含下面的元素：

— <xenc:EncryptedData> [必需的]

加密的内容以及加密的细节，参见 W3C Encryption 中的定义。需要呈现类型属性，如果存在的话，必须包含 http://www.w3.org/2001/04/xmlenc#Element 。加密的内容必须包括一个具有 **AssertionType** 类型或者从中衍生出来的类型的元素。

— <xenc:EncryptedKey> [0 个或者多个]

包装好的解密密钥，参见 W3X Encryption 中的定义。每个包装好的密钥都应该包括一个 Recipient 属性来指定为之加密密钥的实体。Recipient 属性值应该是一个 SAML 系统实体的 URI 标识符，参见第 8.7 节中的定义。

当纯文本值穿过中间媒体的时候，加密断言的目的是作为一个机密保护机制。

下面的 Schema 段定义了<EncryptedAssertion>元素：

```
<element name="EncryptedAssertion" type="saml:EncryptedElementType"/>
```

8.1.4 主体

本部分定义了用来描述一个断言的主体的 SAML 构造。可选的<Subject>元素定义了责任人，它是断言中所有断言（0 个或者多个）的主体。它包括一个标识符，一系列一个或者多个主体确认，或者两个都包括：

— <BaseID>, <NameID>或<EncryptedID> [可选的]

标识一个主体

— <SubjectConfirmation> [0 个或者多个]

允许确认主体的信息。如果提供了多于一个主体确认信息，那么为了应用断言，满足任何一个，对于确认主体来讲就足够了。

<Subject>元素可以包括一个标识符以及 0 个或者多个主体确认信息，当一个可信任方在处理一个断言的时候，可以对这些确认信息进行验证。如何所包含的任何一个主体确认信息得到了验证，那么可信任方可以把提出断言的团体作为一个已经和责任人相关联的断言方，在名字标识符中标识了该责任人，同时该责任人和断言中的断言相关联。证明实体和实际的主体可能是也可能不是同样一个实体。

如果没有包括主体确认信息，那么不规定断言提交者和实际主体之间的关系。

一个<Subject>元素应不标识多于一个责任人。

下面的 Schema 段定义了<Subject>元素及其 **SubjectType** 复杂类型：

```
<element name="Subject" type="saml:SubjectType"/>
<complexType name="SubjectType">
      <choice>
       <sequence>
             <choice>
                   <element ref="saml:BaseID"/>
                   <element ref="saml:NameID"/>
                   <element ref="saml:EncryptedID"/>
             </choice>
             <element ref="saml:SubjectConfirmation" minOccurs="0"
maxOccurs="unbounded"/>
       </sequence>
       <element ref="saml:SubjectConfirmation" maxOccurs="unbounded"/>
      </choice>
</complexType>
```

8.1.4.1 <SubjectConfirmation>元素

<SubjectConfirmation>元素为一个可信任方提供验证断言的主体与正在与信任方通信的一方相互关系的方法。它包括下面的属性和元素：

— Method [必需的]

一个 URI 引用，标识一个用来确认主体的协议或者机制。在第 11 节中定义了用来标识 SAML 定义的确认方法的 URI 引用。可以通过规定新的 URI 和协议子集或者通过私有协议来增加额外的方法。

— <BaseID>, <NameID>或者<EncryptedID> [可选的]

标识一个实体，期望该实体可以满足封装主体确认需求：

— <SubjectConfirmationData> [可选的]

特定的确认方式采用的额外的确认信息。例如，这个元素的典型内容可能是<ds:KeyInfo>元素（参见 W3C Encryption 中的定义），它标识了一个密钥（参见第 8.1.4.3 节）。特定的确认方式可以定义一个 Schema 类型从而描述出现在<SubjectConfirmationData>元素中的元素、属性或者内容。

下面的 Schema 段定义了<SubjectConfirmationData>元素及其 **SubjectConfirmationType** 复杂类型：

```
<element name="SubjectConfirmation" type="saml:SubjectConfirmationType"/>
<complexType name="SubjectConfirmationType">
       <sequence>
              <choice minOccurs="0">
                     <element ref="saml:BaseID"/>
                     <element ref="saml:NameID"/>
                     <element ref="saml:EncryptedID"/>
              </choice>
              <element ref="saml:SubjectConfirmationData" minOccurs="0"/>
       </sequence>
       <attribute name="Method" type="anyURI" use="required"/>
</complexType>
```

8.1.4.2 <SubjectConfirmationData>元素

<SubjectConfirmationData>元素带有 **SubjectConfirmationDataType** 复杂类型。它规定了确认主体的额外数据或者限制了能够发生主体确认的环境。当一个可信任方试图验证提出断言的一个实体（也就是说，证明实体）和断言主体之间关系的时候，会发生主体确认。它包含下面的可以应用到任何方法上的可选属性：

— NotBefore [可选的]

主体被确认之前的一个时间点。通过 UTC 编码时间值，参见第 7.3 节中的描述。

— NotOnOrAfter [可选的]

不能在继续确认主体的一个时间点。通过 UTC 编码时间值，参见第 7.3 节中的描述。

— Recipient [可选的]

一个定义了实体或者位置的 URI，一个证明实体可以向该实体提出断言。例如，这个属性可以指示必须通过一个特定的网络端点传递断言以防止中间节点把它发送到另外的位置。

— InResponseTo [可选的]

一个 SAML 协议消息的 ID，用来响应可以提出断言的一个证明实体。例如，可以利用这个属性来把断言和一个导致这个属性出现的 SAML 请求关联起来。

— Address [可选的]

一个证明实体可以提出断言的网络地址/位置。例如，可以利用这个属性把断言和特定的客户地址映射起来从而防止攻击者的早期偷窃并且防止提交来自其他位置的断言。应该利用常用的点分十进制格式（例如，“1.2.3.4”）来标识 IPv4 地址。利用 IETF RFC3513，第 2.2 节中定义的方式来表示 IPv6 地址（例如，"FEDC:BA98:7654:3210:FEDC:BA98:7654:3210"）。

— 任意属性

本复杂类型通过<xs:anyAttribute>扩展点允许为<SubjectConfirmationData>结构增加任意的符合命名空间要求的 XML 属性，而无需显式的 Schema 扩展。这需要增加额外的字段以支持和确认有关的信息。SAML 扩展决不能通过 SAML 定义的命名空间为 **SubjectConfirmationDataType** 复杂类型增加本地（不符合命名空间需求的）XML 属性或者满足 XML 属性的属性或者是通过 XML 属性衍生出来的属性；这些属性用于未来 SAML 本身的维护和改进。

— 任意元素

这个复杂类型利用一个<xs:any>扩展点允许为<SubjectConfirmationData>结构增加任意的 XML 元素，而无需显式的 Schema 扩展。因此在需要提供额外的与确认有关的信息的时候，可以增加额外的元素。

利用这些方式的特定的确认方法和协议子集可能会用到在这个复杂类型中定义的一个或者多个属性。如何利用这些属性（以及通用的主体确认）的例子，参见第 13 节。

如果出现了，由可选的 NotBefore 和 NotOnOrAfter 定义的时间周期，那么这个时间周期应该在<Conditions>元素的 NotBefore 和 NotOnOrAfter 属性定义的整个断言验证周期内。如果两种属性都出现了，那么 NotBefore 的值必须小于（早于）NotOnOrAfter 的值。

下面的 Schema 段定义了<SubjectConfirmationData>元素及其 **SubjectConfirmationDataType** 复杂类型：

```
<element name="SubjectConfirmationData"
type="saml:SubjectConfirmationDataType"/>
<complexType name="SubjectConfirmationDataType" mixed="true">
      <complexContent>
            <restriction base="anyType">
                  <sequence>
                        <any namespace="##any" processContents="lax"
minOccurs="0" maxOccurs="unbounded"/>
                  </sequence>
                  <attribute name="NotBefore" type="dateTime"
use="optional"/>
                  <attribute name="NotOnOrAfter" type="dateTime"
use="optional"/>
                  <attribute name="Recipient" type="anyURI"
use="optional"/>
                  <attribute name="InResponseTo" type="NCName"
use="optional"/>
                  <attribute name="Address" type="string"
use="optional"/>
                  <anyAttribute namespace="##other"
processContents="lax"/>
            </restriction>
      </complexContent>
</complexType>
```

8.1.4.3 复杂类型KeyInfoConfirmationDataType

KeyInfoConfirmationDataType 复杂类型限制一个<SubjectConfirmationData>元素要包含一个或者多个<ds:KeyInfo>元素，用来标识通过某种方式验证一个认证实体的密钥。特定的确认方式必须定义使用确认数据的明确的方式。也可能出现由 **SubjectConfirmationDataType** 复杂类型定义的可选属性。

任何一个确认方式都可以采用这个复杂类型或者从它衍生出来的一个类型，这些确认方式根据<ds:KeyInfo>元素定义了该类型的确认数据。

根据 W3C Encryption，每个<ds:KeyInfo>元素必须标识一个单独的密钥。多个密钥应该通过独立的<ds:KeyInfo>元素来标识，例如一个责任人利用不同的密钥向不同的可信任方确认自己。

下面的 Schema 段定义了 **KeyInfoConfirmationDataType** 复杂类型：

```
<complexType name="KeyInfoConfirmationDataType" mixed="false">
       <complexContent>
              <restriction base="saml:SubjectConfirmationDataType">
                     <sequence>
                            <element ref="ds:KeyInfo" maxOccurs="unbounded"/>
                     </sequence>
              </restriction>
       </complexContent>
</complexType>
```

8.1.4.4 确认密钥的<Subject>的实例

为了说明不同的元素和类型之间如何组合在一起，下面给出了一个实例，在这个实例中一个<Subject>元素包含一个名字标识符以及一个基于拥有一个密钥的主体确认信息。此外，利用 **KeyInfoConfirmationDataType** 来标识确认数据句法的是一个<ds:KeyInfo>元素：

```
<Subject>
       <NameID
Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
       scott@example.org
       </NameID>
       <SubjectConfirmation
Method="urn:oasis:names:tc:SAML:2.0:cm:holder-of-key">
              <SubjectConfirmationData
xsi:type="saml:KeyInfoConfirmationDataType">
                     <ds:KeyInfo>
                            <ds:KeyName>Scott's Key</ds:KeyName>
                     </ds:KeyInfo>
              </SubjectConfirmationData>
       </SubjectConfirmation>
</Subject>
```

8.1.5 Conditions条件

本部分定义了限制 SAML 断言可接受的 SAML 结构。<Conditions>元素可能包含下面的元素和属性：

— NotBefore [可选的]

定义了断言有效的最早的时间点。利用 UTC 来编码时间值，参见第 7.3 节中的描述。

— NotOnOrAfter [可选的]

定义了断言到期的时间点。利用 UTC 来编码时间值，参见第 7.3 节中的描述。

— <Condition> [任意数量]

在扩展 Schema 中定义的一个类型的条件。必须利用一个 xsi:type 属性来说明实际的条件类型。

— <AudienceRestriction> [任意数量]

说明断言指向特定的用户。

— <OneTimeUse> [可选的]

说明应该立即使用断言而且不能留做将来使用。虽然 Schema 允许多次出现，但是最多必须只能有该元素的一个实例。

— <ProxyRestriction> [可选的]

定义了断言实体对希望自己成为断言实体并且根据包含在最初断言内的信息发布自己断言的可靠方的限制条件。虽然 Schema 允许多次出现，但最多必须只能有该元素的一个实例。

因为采用 xsi:type 属性允许一个断言包含多于一个的 **ConditionsType**（例如 **OneTimeUseType**）的 SAML 定义的子类型，Schema 不会明确地规定包含特定条件的次数。特定的条件类型会定义对于这种使用情况的限制，参见上面的描述。

下面的 Schema 段定义了<Conditions>元素及其 **ConditionsType** 复杂类型：

```
<element name="Conditions" type="saml:ConditionsType"/>
<complexType name="ConditionsType">
      <choice minOccurs="0" maxOccurs="unbounded">
            <element ref="saml:Condition"/>
            <element ref="saml:AudienceRestriction"/>
            <element ref="saml:OneTimeUse"/>
            <element ref="saml:ProxyRestriction"/>
      </choice>
      <attribute name="NotBefore" type="dateTime" use="optional"/>
      <attribute name="NotOnOrAfter" type="dateTime" use="optional"/>
</complexType>
```

8.1.5.1 通用处理规则

如果断言包含一个<Conditions>元素，那么利用下面按次序给出的规则，根据提供的子元素和属性来对断言进行验证。

但是一个具有条件验证状态 Valid 的断言可能是不可靠的或者无效的，原因可能是它的格式不对或者非 Schema 有效的，不是由一个可信赖的 SAML 授权机构签发的，或者是没有通过一个可信任方式进行认证。

有些条件可能不会直接影响到所包含断言的有效性（它们总是被评估为有效），但是可能会限制到可信任方使用断言的行为。

— 如果在<Conditions>元素中没有提供子元素或者属性，那么根据对条件的处理，认为断言有效。

— 如果<Conditions>元素的任何一个子元素或者属性都无效，那么断言无效。

— 如果无法对<Conditions>元素的任何一个子元素或者属性进行评估，或者无法理解遇到的元素，那么无法决定断言的有效性并且认定它为Indeterminate。

— 如果认定<Conditions>元素的子元素和属性有效，那么根据条件的处理程序，认为断言有效。

应用的第一个规则终止条件处理；那么认定一个断言无效的决定会优先于 Indeterminate。

一个可信任方（无论它正在处理什么关联或协议子集）必须拒绝无效或者不确定的断言，和断言畸形或者不可用的情况一样。

8.1.5.2 NotBefore 和 NotOnOrAfter属性

NotBefore 和 NotOnOrAfter 属性在协议子集的使用关联中针对断言的有效性定义了时间限制。在断言的有效期内它们无法保证断言中断言的正确或者精确。

NotBefore 属性定义了有效性间隔开始的时间点。NotOnOrAfter 属性定义了有效性间隔结束的时间点。

在省略了 NotBefore 或者 NotOnOrAfter 的值的情况下，那么没有定义属性（如果提供的所有其他条件都被评估为有效），那么在 NotOnOrAfter 属性定义的时间点之前的任何时间的条件下，断言都有效。如果没有定义 NotOnOrAfter 属性（如果提供的所有其他条件都被评估为有效），那么根据在 NotBefore 属性定义的时间点没有过期的情况下，断言有效。如果没有定义这两个属性（如果提供的其他条件都被评估为有效），那么任何时间内的断言都有效。

如果出现了两个属性，那么 NotBefore 的值务必小于（早于）NotOnOrAfter 的值。

8.1.5.3 <Condition>元素

<Condition>元素作为新条件的扩展点。它的 **ConditionAbstractType** 复杂类型是摘要性质的，因此只能用做一个衍生类型的基础。

下面的 Schema 段定义了<Condition>元素及其 **ConditionAbstractType** 复杂类型：

```
<element name="Condition" type="saml:ConditionAbstractType"/>
<complexType name="ConditionAbstractType" abstract="true"/>
```

8.1.5.4 <AudienceRestriction>和<Audience>元素

<AudienceRestriction>元素规定断言指向由<Audience>元素标识的一个或者多个特定的用户。虽然一个在指定的用户范围外的 SAML 可信任方可以从一个断言得出结论，但是 SAML 断言方明显没有把这样一个团体作为精确的或者值得信任的团体。它包括下面的元素：

— <Audience>

标识一个目的用户的 URI 引用。URI 引用可以标识一个描述术语和用户成员情况的文件。它可能还包括来自一个描述一个系统实体的 SAML 名字标识符的唯一的标识符 URI。

当且仅当 SAML 可信任方是规定用户成员中的一个，那么用户限制条件评估为有效。

SAML 断言方无法阻止一方根据暴露给它的断言的信息来采取一定的动作。但是，<AudienceRestriction>元素允许 SAML 断言方明确地断言没有通过机器可读或者人类可读的格式来为这样的一个团体提供保证。虽然不能保证在所有的环境下都支持这样一个免担保属性，但是支持免担保的概率已经有了相当程度的提高。

在单个断言中可能包含多个<AudienceRestriction>，必须独立评估每个断言。该需求以及前面定义的作用就是，在一个给定条件下，用户形成一个分离（“OR”）的关系，同时多个条件形成一个联合（“AND”）的关系。

接下来的 Schema 段定义了<AudienceRestriction>属性及其 **AudienceRestrictionType** 复杂类型：

```
<element name="AudienceRestriction"
       type="saml:AudienceRestrictionType"/>
<complexType name="AudienceRestrictionType">
       <complexContent>
              <extension base="saml:ConditionAbstractType">
                     <sequence>
                            <element ref="saml:Audience"
maxOccurs="unbounded"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
<element name="Audience" type="anyURI"/>
```

8.1.5.5 <OneTimeUse>元素

通常，可信任方可能会选择以某种其他的格式保留它们包含的断言或者信息，以备重新使用。<OneTimeUse>条件元素允许一个授权机构说明断言中的信息可能很快就会发生变化，因此应该为每次使用获得新信息。一个实例就是，包含一个<AuthzDecisionStatement>的断言,该断言是一个策略的结果，该策略规定作为日期时间的一个函数的接入控制。

如果在分布式环境下，系统始终可以精确地同步，那么如果仔细地使用有效性间隔，可能会满足这个要求。但是，因为在系统之间，时钟通常会滑动并且可能和传输时延结合起来，因此对于发起者来讲，没有方便的办法来适当地限制一个断言的生存期，因为有可能在断言还没有到达的时候，它就已经过期了。

<OneTimeUse>元素指示可信任方应该立即使用断言而且不能保留以备将来使用。通常在每次使用的时候，可信任方可以自由地请求一个新的断言。但是，选择保留断言以备将来使用的执行情况必须遵守<OneTimeUse>元素的规定。这个条件和 NotBefore 以及 NotOnOrAfter 条件信息无关。

为了支持单次使用的限制，一个可信任方应该维护一个它已经处理过的包含这样一个条件的断言的缓存。无论何时处理具有这种条件的断言，应该检查缓存器来确保先前没有收到过这个断言而且该断言没有被可信任方处理过。

一个 SAML 授权机构在一个断言的<Conditions>元素内不能包括多于一个<OneTimeUse>元素。

出于决定<Conditions>元素有效性的目的，通常认为<OneTimeUse>有效。也就是说，这个条件只是一个在使用的条件，不会影响有效性。

下面的 Schema 段定义了<OneTimeUse>元素及其 **OneTimeUseType** 复杂类型：

```
<element name="OneTimeUse" type="saml:OneTimeUseType"/>
<complexType name="OneTimeUseType">
       <complexContent>
              <extension base="saml:ConditionAbstractType"/>
</complexContent>
</complexType>
```

8.1.5.6 <ProxyRestriction>元素

定义了断言方强加到可信任方上的限制条件，这些信赖方希望可以作为可断言方并且根据包含在起始断言内的信息发布接下来的自己的断言。根据包含这样一个条件的断言，作为断言方的可信任方不能发布会影响到本条件中定义的限制条件的断言。

<ProxyRestriction>元素包含下面的元素和属性：

— Count [可选的]

定义了断言方允许在这个断言和根据它最终签发的断言之间断言方允许存在的间接最大数量。

— <Audience> [0 个或者多个]

定义了用户集，断言方允许根据这个断言为这些用户签发新断言。

0 个计数值，它指示了可信任方不必根据这个断言签发到另外一个可信任方的断言。如果大于 0，那么任何签发的断言，它们自己必须包含一个<ProxyRestriction>元素，而且计数值最多比这个值小 1。

如果没有定义<Audience>元素，那么就没有为需要给之签发接下来断言的可信任方强加用户限制。否则，任何签发的断言必须本身包含一个<AudienceRestriction>元素而且在之前的<ProxyRestriction>元素中至少出现了一个<Audience>元素，同时没有出现在之前<ProxyRestriction>元素内没有出现的<Audience>元素。

在一个断言的<Conditions>元素内，一个 SAML 授权机构不能包括多于一个<ProxyRestriction>元素。

出于决定<Conditions>元素有效性的目的，通常认为<ProxyRestriction>条件有效。也就是说，这个条件不会影响有效性，只是一个正在使用的条件。

下面的 Schema 段定义了<ProxyRestriction>元素及其 **ProxyRestrictionType** 复杂类型：

```
<element name="ProxyRestriction" type="saml:ProxyRestrictionType"/>
<complexType name="ProxyRestrictionType">
       <complexContent>
              <extension base="saml:ConditionAbstractType">
                     <sequence>
                            <element ref="saml:Audience" minOccurs="0"
maxOccurs="unbounded"/>
                     </sequence>
                     <attribute name="Count" type="nonNegativeInteger"
use="optional"/>
```

```
            </extension>
      </complexContent>
</complexType>
```

8.1.6 建议

本部分定义了包含关于一个断言方希望提供给一个可信任方的断言的附加信息的 SAML 结构。

<Advice>元素包含了一个 SAML 授权机构希望提供的任何附加信息。在不影响断言的语义或者有效性的情况下，应用可能忽略这些信息。

<Advice>元素包含 0 个或者多个<Assertion>、<EncryptedAssertion>、<AssertionIDRef>、<AssertionURIRef>元素以及在其他非 SAML 命名空间中有效命名空间的元素的集合。

接下来介绍了<Advice>元素的一些可能的应用：

- 包括支持引用断言需求的证据，直接（通过合并断言）或者间接的（通过引用支持的断言）。
- 给出一个断言应用的证据。
- 定义了更新断言的定时和分布点。

下面的 Schema 段定义了<Advice>元素及其 **AdviceType** 复杂类型：

```
<element name="Advice" type="saml:AdviceType"/>
<complexType name="AdviceType">
      <choice minOccurs="0" maxOccurs="unbounded">
            <element ref="saml:AssertionIDRef"/>
            <element ref="saml:AssertionURIRef"/>
            <element ref="saml:Assertion"/>
            <element ref="saml:EncryptedAssertion"/>
            <any namespace="##other" processContents="lax"/>
      </choice>
</complexType>
```

8.1.7 断言

所有 SAML 定义的断言都和一个主体相关联。通常把 SAML 断言均是针对由<Subject>元素来表示一个主体。但是，<Subject>元素为可选，其他的规范或者协议子集可能利用 SAML 断言结构来生成类似的断言但是不规定一个主体，或者可能通过可替代的方法来规定主体。接下来的各节定义了包含了这些断言信息的 SAML 结构。

8.1.7.1 <Statement>元素

<Statement>元素是一个扩展点，该扩展点允许其他的基于断言的应用重新利用 SAML 断言框架。SAML 本身来源于这个扩展点的核心断言。它的 **StatementAbstractType** 复杂类型是概要性质的，因此只能作为衍生类型的基础。

下面的 Schema 段定义了<Statement>元素及其 **StatementAbstractType** 复杂类型：

```
<element name="Statement" type="saml:StatementAbstractType"/>
<complexType name="StatementAbstractType" abstract="true"/>
```

8.1.7.2 <AuthnStatement>元素

<AuthnStatement>元素由宣称断言主体在某特定的时刻通过特定的方法认证了断言主体的 SAML 授权机构描述的一个声明。包含<AuthnStatement>元素的断言务必包含一个<Subject>元素。

这是一个 **AuthnStatementType** 类型，它通过增加下面的元素和属性扩展了 **StatementAbstractType**。

注 — 从SAML V2.0的<AuthnStatement>去掉了<AuthorityBinding>元素以及与它相应的类型。

— AuthnInstant [必需的]

定义了认证发生的时间。通过 UTC 编码时间值，参见第 7.3 节中的描述。

— SessionIndex [可选的]

定义了由主体标识的责任人和认证过的授权机构之间的一个特定会话索引。

— SessionNotOnOrAfter [可选的]

定义了一个时间点，必须认定在这个时间点上通过主体和发起这个断言的 SAML 授权机构标识的责任人之间的会话已经停止。通过 UTC 编码时间值，参见第 7.3 节中的描述。在这个属性以及可能出现在这个断言中的 NotOnOrAfter 条件属性之间没有必备的关系。

— <SubjectLocality> [可选的]

为系统定义了 DNS 域名和 IP 地址，通过该系统，明显地认证了断言主体。

— <AuthnContext> [必需的]

认证授权机构采用的关联，包括产生这个断言的认证事件。包含认证关联类别的参考、一个认证关联断言或者断言参考或者两个都包括。需要了解有关认证关联信息的全部内容参见第 12 节（认证关联）。

通常，任何一个字符串值都可以用做一个 SessionIndex 值。但是，在考虑到保密问题时，必须保证 SessionIndex 值不会使其他的保密机制失效。因此，责任人不能利用该值来在不同的会话参与者之间来关联行为。接下来提供了两个可以获得这个目标的解决方案并且建议采用这两种方案：

- 采用小的正整数（或者列表中的再现常数）用于SessionIndex。SAML授权机构应该选择值的范围，因此任何一个整数的集的势需要足够高以阻止一个特定责任人的行为在多个会话参与者之间被关联。SAML权威机构应该在这个范围内为SessionIndex随机选值（除非需要保证给予同一个会话参与者的后续断言的值的唯一性，这个会话参与者作为另外一个会话的一部分）。
- 在SessionIndex中采用了封装的断言ID值。

下面的 Schema 段定义了<AuthnStatement>元素及其 **AuthnStatementType** 复杂类型：

```
<element name="AuthnStatement" type="saml:AuthnStatementType"/>
<complexType name="AuthnStatementType">
       <complexContent>
              <extension base="saml:StatementAbstractType">
                     <sequence>
                            <element ref="saml:SubjectLocality"
minOccurs="0"/>
                            <element ref="saml:AuthnContext"/>
                     </sequence>
                     <attribute name="AuthnInstant" type="dateTime"
use="required"/>
                     <attribute name="SessionIndex" type="string"
use="optional"/>
                     <attribute name="SessionNotOnOrAfter" type="dateTime"
use="optional"/>
              </extension>
       </complexContent>
</complexType>
```

8.1.7.2.1 <SubjectLocality>元素

<SubjectLocality>元素为认证了断言主体的系统定义了 DNS 域名和 IP 地址。它具有下面的属性：

— Address [可选的]

系统的网络地址，通过该系统，由主体标识的责任人得到了认证。IPv4 地址应该用点分十进制格式表示（例如，“1.2.3.4”）。IPv6 地址应该用 IETF RFC3513 第 2.2 节中的定义来表示（例如，"FEDC:BA98:7654:3210:FEDC:BA98:7654:3210"）。

— DNSName [可选的]

系统的 DNS 名字，通过这个系统，由主体标识的责任人得到了认证。

因为这个元素的这些字段都很容易受到“哄骗”，因此，这个元素完全就是建议性质的，但是在某些应用中它可能是有用的信息。

下面的 Schema 段定义了<SubjectLocality>元素及其 **SubjectLocalityType** 复杂类型：

```
<element name="SubjectLocality" type="saml:SubjectLocalityType"/>
<complexType name="SubjectLocalityType">
      <attribute name="Address" type="string" use="optional"/>
      <attribute name="DNSName" type="string" use="optional"/>
</complexType>
```

8.1.7.2.2 <AuthnContext>元素

<AuthnContext>元素定义了一个认证事件的关联。该元素可以包含一个认证关联类别引用、一个认证关联断言或者一个断言引用或者两个都包括。它的复杂类型 **AuthnContextType** 具有下面的元素：

— <AuthnContextClassRef> [可选的]

一个标识了认证关联类别的 URI 引用，该类别描述了后随的认证关联断言：

— <AuthnContextDecl>或者<AuthnContextDeclRef> [可选的]

可以是值提供的一个认证关联断言，或者是标识这样一个断言的 URI 引用。URI 引用可以直接分解为一个包含被引用断言的 XML 文档。

— <AuthenticatingAuthority> [0 个或者多个]

责任人认证涉及的认证授权机构的 0 个或者多个唯一标识符（没有包括断言发起者，假定涉及了断言发起者，但是在此处没有明确指出）。

需要了解认证关联信息的全部内容，参见第 12 节。

下面的 Schema 段定义了<AuthnContext>元素及其 **AuthnContextType** 复杂类型：

```
<element name="AuthnContext" type="saml:AuthnContextType"/>
<complexType name="AuthnContextType">
      <sequence>
            <choice>
                  <sequence>
                        <element ref="saml:AuthnContextClassRef"/>
                        <choice minOccurs="0">
                              <element ref="saml:AuthnContextDecl"/>
                              <element ref="saml:AuthnContextDeclRef"/>
                        </choice>
                  </sequence>
                  <choice>
                        <element ref="saml:AuthnContextDecl"/>
                        <element ref="saml:AuthnContextDeclRef"/>
                  </choice>
            </choice>
            <element ref="saml:AuthenticatingAuthority" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
</complexType>
<element name="AuthnContextClassRef" type="anyURI"/>
<element name="AuthnContextDeclRef" type="anyURI"/>
<element name="AuthnContextDecl" type="anyType"/>
<element name="AuthenticatingAuthority" type="anyURI"/>
```

8.1.7.3 <AttributeStatement>元素

<AttributeStatement>元素描述了由 SAML 权威机构断言的一个断言，它说明断言主体和特定的属性有关。包含<AttributeStatement>元素的断言必须包含一个<Subject>元素。

它是 **AttributeStatementType**，它通过附加下面的元素扩展了 **StatementAbstractType**：

— <Attribute>或者<EncryptedAttribute> [1 个或者多个]

<Attribute>元素规定断言主体的一个属性。一个加密的 SAML 属性可能包括在<EncryptedAttribute>元素中：

下面的 Schema 段定义了<AttributeStatement>元素及其 **AttributeStatementType** 复杂类型：

```
<element name="AttributeStatement" type="saml:AttributeStatementType"/>
<complexType name="AttributeStatementType">
       <complexContent>
              <extension base="saml:StatementAbstractType">
                     <choice maxOccurs="unbounded">
                            <element ref="saml:Attribute"/>
                            <element ref="saml:EncryptedAttribute"/>
                     </choice>
              </extension>
       </complexContent>
</complexType>
```

8.1.7.3.1 <Attribute>元素

<Attribute>元素利用名字定义了一个属性，有可能有选择性地加上它的值。它有 **AttributeType** 复杂类型。它用在一个属性断言内，解释和一个断言主体有关的特定的属性和值，参见前一个节的描述。它也可以用在一个属性中，用来询问特定 SAML 属性值的返回。<Attribute>元素包含下面的 XML 属性：

— Name [必需的]

属性的名称。

— NameFormat [可选的]

这是一个 URI 引用，描述了属性名字的类别，用于解释名字的含义。需要了解可能用做 NameFormat 属性的值的一些 URI 应用，它们相关的描述和处理规则参见第 8.7.2 节。如果没有提供 NameFormat 值，标识符 urn:oasis:names:tc:SAML:2.0:attrname-format:unspecifie 有效。

— FriendlyName [可选的]

一个提供了人类更容易读懂的格式的属性名称的字符串，当实际名称比较复杂或者不透明的时候，它比较有用，例如一个 OID（参见 ITU-T X.660 建议书中的定义）或者一个 UUID（参见 ITU-T X.667 建议书中的定义）。这个属性值不能作为正式标识 SAML 属性的基础。

— 任意属性

这个复杂类型采用一个<xs:anyAttribute>扩展点来允许将任意的 XML 属性添加到<Attribute>结构中而无需一个显式的 Schema 扩展。这就使得在需要的时候可以增加额外的字段来提供所需的额外参数，例如，一个属性查询中，SAML 扩展不能为 **AttributeType** 复杂类型或者从它衍生出来的类型增加本地（不符合命名空间的）的 XML 属性或者符合一个 SAML 定义的命名空间的 XML 属性；这些属性预留用于将来 SAML 本身的维护和改进。

— <AttributeValue> [任意数量]

包含属性的一个值。如果一个属性包含多个不连续值，建议每个值出现在它自身的<AttributeValue>元素中。如果为一个属性提供了多个<AttributeValue>元素，而且每个元素都具有一个 xsi:type 分配的 datatype，那么所有的<AttributeValue>元素必须具有指配的相同 datatype。

没有包含<AttributeValue>元素的<Attribute>元素的意义取决于和它的关联。在<AttributeStatement>中，如果 SAML 属性存在但是没有值，那么必须省略<AttributeValue>元素。在<samlp:AttributeQuery>中，值的缺少暗示着请求端对任何一个或者全部指定的属性值感兴趣（参见第 8.2 节的介绍）。

协议子集或者其他规范如果想使用<Attribute>元素必须定义指定或者省略<AttributeValue>元素的语义。

接下来的 Schema 段定义了<Attribute>元素及其 **AttributeType** 复杂类型：

```
<element name="Attribute" type="saml:AttributeType"/>
<complexType name="AttributeType">
       <sequence>
              <element ref="saml:AttributeValue" minOccurs="0"
maxOccurs="unbounded"/>
       </sequence>
       <attribute name="Name" type="string" use="required"/>
       <attribute name="NameFormat" type="anyURI" use="optional"/>
       <attribute name="FriendlyName" type="string" use="optional"/>
       <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
```

<AttributeValue>元素为一个特定的 SAML 属性提供了一个值。它是 **xs:anyType** 类型，它允许任何形式完好的 XML 作为元素的内容出现。

如果一个<AttributeValue>元素的数据内容是一个 XML Schema 简单类型（例如 **xs:integer** 或者 **xs:string**），应该通过<AttributeValue>元素中一个 xsi:type 断言的方式来明确地断言 datatype。如果属性包含结构化数据，可以通过一个扩展 Schema 来定义必须的数据元素。

注 — 为了继续进行 Schema处理过程，利用xsi:type对<AttributeValue>定义一个datatype而不是一个XMLSchema简单类型，要求出现定义datatype的扩展Schema。

如果一个 SAML 属性包括一个空值，例如空的字符串，那么相应的<AttributeValue>元素必须为空（通常被连续化为<AttributeValue/>）。此时没有考虑第 7.1 节中规定的要求，也就是说 SAML 内容中的字符串值至少要包含一个非空字符。

如果一个 SAML 属性包括一个“Null”值，相应的<AttributeValue>元素必须为空而且必须包含预留的 xsi:nil XML 属性，值为“true”或者“1”。

接下来的 Schema 段定义了<AttributeValue>元素：

```
<element name="AttributeValue" type="anyType" nillable="true"/>
```

8.1.7.3.2 <EncryptedAttribute>元素

<EncryptedAttribute>元素利用加密方式表示了一个 SAML 属性，参见 W3C Encryption 中的定义。<EncryptedAttribute>元素包含下面的元素：

— <xenc:EncryptedData> [必备]

加密的内容以及相关的加密细节，参见 W3C Encryption 中的定义。Type 属性应出现，若出现类型属性，必须包含一个 http://www.w3.org/2001/04/xmlenc#Element 值。加密的内容必须包含一个具有 **AttributeType** 类型或者从它衍生出来的类型的元素。

— <xenc:EncryptedKey> [0 个或者多个]

封装的解密密钥，参见 W3C Encryption 中的定义。每个封装的密钥应该包括一个 Recipient 属性，它定义了为之加密密钥的实体。Recipient 属性值应该是具有一个 SAML 名字标识符的一个系统实体的 URI 标识符，参见第 8.7 节。

当纯文本值穿越中间媒介的时候，加密的属性可以作为一种机密性保护机制。

接下来的 Schema 段定义了<EncryptedAttribute>元素：

```
<element name="EncryptedAttribute" type="saml:EncryptedElementType"/>
```

8.1.7.4 <AuthzDecisionStatement>元素

<AuthzDecisionStatement>元素描述了 SAML 授权机构的一个断言，断言通过断言主体访问特定资源的请求已经根据一些可选的指定条件得出了一个详细的授权决定。包括<AuthzDecisionStatement>元素的断言必须包含一个<Subject>元素。

通过一个 URI 引用来标识资源。为了能够将断言解释得正确而且安全，SAML 授权机构和 SAML 可信任方必须采用一致的方式来解释每个 URI 引用。根据对资源 URI 引用的编码，如果没有采用一致的 URI 引用解释方式可能会导致不同的授权决定。标准化 URI 引用的规则可以参见 IETF RFC2396 第 6 节。

为了避免由于 URI 编码的变异而导致的不明确，SAML 系统实体应该尽可能地采用 URI 标准化格式：

- SAML授权机构应该利用标准化格式来编码所有的资源URI引用。
- 在处理之前，可信任方将资源URI参数转化为标准形式。

可信任方在处理资源 URI 引用时，应该先把它们转换到标准化格式。URI 引用句法和一个底层文件系统语义之间的区别也会导致不一致的 URI 引用解释。如果采用 URI 引用来定义接入控制策略语言，那么需要特别注意。采用 SAML 断言的系统应该满足下面的安全条件：

- 部分URI引用句法需要区分大小写。如果底层的文件系统需要区分大小写，此时如果改变请求端一部分资源URI引用的大小写，那么应该无法访问被拒绝的资源。
- 许多文件系统支持的机制，例如逻辑通路和符号链路，允许用户在文件系统条目之间建立逻辑对等。但是一个请求端应该不能通过建立这样的对等来访问一个被拒绝的资源。

<AuthzDecisionStatement>元素是一个 **AuthzDecisionStatementType** 类型，它通过附加下面的元素和属性扩展了 **StatementAbstractType**：

— Resource[必需的]

标识了所需资源的 URI 引用，此时，需要得到到该资源的接入授权。这个属性可能具有空的 URI 引用（“”），它的含义是“当前文档的起点”，参见 IETF RFC2396,第 4.2 节中的定义。

— Decision [必备]

SAML 授权机构根据特定的资源提交的结果。值是 **DecisionType** 简单类型。

— <Action> [1 个或者多个]

得到授权可以在特定资源上执行的行为集。

— <Evidence> [可选的]

SAML 授权机构在做决定时所依赖的断言集。

接下来的 Schema 段定义了<AuthzDecisionStatement>元素及其 **AuthzDecisionStatementType** 复杂类型：

```
<element name="AuthzDecisionStatement"
 type="saml:AuthzDecisionStatementType"/>
<complexType name="AuthzDecisionStatementType">
 <complexContent>
  <extension base="saml:StatementAbstractType">
        <sequence>
              <element ref="saml:Action" maxOccurs="unbounded"/>
              <element ref="saml:Evidence" minOccurs="0"/>
        </sequence>
        <attribute name="Resource" type="anyURI" use="required"/>
        <attribute name="Decision" type="saml:DecisionType" use="required"/>
  </extension>
 </complexContent>
</complexType>
```

8.1.7.4.1 DecisionType简单类型

DecisionType 简单类型定义了将要报告的可能的值，它作为一个授权决定断言的状态。

— 允许

允许特定的行为。

— 拒绝

拒绝特定的行为。

— 不确定

SAML 授权机构无法决定接收还是拒绝特定的行为。

`Indeterminate` 决定值用在授权机构需要提供一个肯定状态的能力但是此时它无法发布一个决定的情况下。至于拒绝或者无法提供一个决定的额外信息可能会在封装的`<Response>`中作为`<StatusDetail>`元素返回。

接下来的 Schema 段定义了 **DecisionType** 简单类型：

```
<simpleType name="DecisionType">
      <restriction base="string">
            <enumeration value="Permit"/>
            <enumeration value="Deny"/>
            <enumeration value="Indeterminate"/>
      </restriction>
</simpleType>
```

8.1.7.4.2 <Action>元素

`<Action>`元素定义了一个针对特定资源的需要得到允许的行为。其中数据内容为将要在特定资源上执行的行为提供了一个标签，它具有下面的属性：

— `Namespace` [可选的]

描述了命名空间的一个 URI 引用，在这个命名空间中，将要解释特定行为的名字。如果没有这个元素，那么在第 8.7 节中定义的命名空间 `urn:oasis:names:tc:SAML:1.0:action:rwedc-negation` 有效。

注（资料性的）— PE36（参见OASIS PE:2006）建议利用下面的内容替代上面的内容：

命名空间[必备]

描述了命名空间的一个URI引用，在这个命名空间中，将要解释特定行为的名字。

接下来的 Schema 段定义了`<Action>`元素及其 **ActionType** 复杂类型：

```
<element name="Action" type="saml:ActionType"/>
<complexType name="ActionType">
 <simpleContent>
       <extension base="string">
             <attribute name="Namespace" type="anyURI" use="required"/>
       </extension>
 </simpleContent>
</complexType>
```

8.1.7.4.3 <Evidence>元素

`<Evidence>`元素包含一个或者多个断言或者断言引用，SAML 授权机构需要依赖它们来发布授权决定。它具有 **EvidenceType** 类型。它包含了下面的一个或者多个元素：

— `<AssertionIDRef>` [任意数量]

它通过引用断言的 ID 属性值来定义一个断言。

— `<AssertionURIRef>` [任意数量]

通过一个 URI 引用来定义一个断言。

— `<Assertion>` [任意数量]

通过值定义一个断言。

— `<EncryptedAssertion>` [任意数量]

通过值定义一个加密的断言。

利用一个断言作为证据可能会影响到 SAML 可信任方之间的可靠协议和 SAML 授权机构所作的授权决定。例如，如果 SAML 可信任方通过一个请求向 SAML 授权机构提交了一个断言，SAML 授权机构可能会利用这个断言作为一个证据来做出授权决定而不会认可`<Evidence>`元素的断言为有效，无论是对可信任方或者是任何一个第三方团体。

接下来的 Schema 段定义了`<Evidence>`元素及其 **EvidenceType** 复杂类型：

```
<element name="Evidence" type="saml:EvidenceType"/>
<complexType name="EvidenceType">
       <choice maxOccurs="unbounded">
              <element ref="saml:AssertionIDRef"/>
              <element ref="saml:AssertionURIRef"/>
              <element ref="saml:Assertion"/>
              <element ref="saml:EncryptedAssertion"/>
       </choice>
</complexType>
```

8.2 SAML协议

可以通过一系列的协议来产生和交换 SAML 协议消息。第 10 节中描述的 SAML 绑定描述了通过现存的广泛部署的传输协议传递协议消息的特定的方法。第 11 节的 SAML 协议子集描述了本节中定义的协议的一系列应用并且描述了执行互通所需的额外的处理规则、限制和需求。

特定的 SAML 请求和应答消息来自通用的类型。请求端向 SAML 响应端发送衍生自 **RequestAbstractType** 的一个元素，响应端生成一个符合或者衍生自 **StatusResponseType** 的一个元素，如图 8-1 所示。

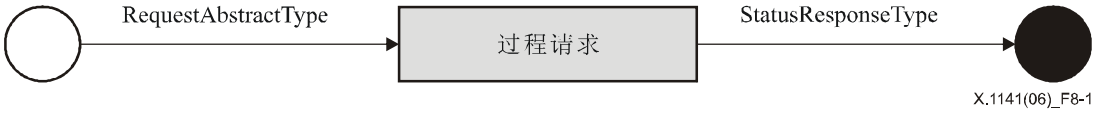

图 8-1/X.1141－SAML 请求 — 应答协议

在某些情况下，如果协议子集允许，在响应端没有接收到相应的请求的时候，也可能生成并且发送一个 SAML 应答。

SAML 定义的协议会实现下面的行为：

— 返回一个或者多个请求的断言。它可以应答一个对于特定断言的直接请求或者是应答对于满足一定条件的断言的询问；

— 根据请求执行认证并且返回相应的断言；

— 根据请求注册一个名字标识符或者终止一个名字的注册；

— 通过凭证机制，来重新获得已经请求的一个协议消息；

— 根据请求对一个有关会话的集合执行一个准同步的注销操作（“单点注销”）；

— 提供一个映射到请求的名字标识符。

在本部分中，在 SAML 协议命名空间的元素和类型的文本描述没有加入传统的命名空间前缀 `samlp:` 。为了清晰起见，通过传统的命名空间 `saml:`来标识 SAML 断言命名空间中关于元素和类型的文本描述。

8.2.1 Schema头和命名空间断言

接下来的 Schema 段为协议 Schema 定义了 XML 命名空间以及其他头信息：

```
<schema
    targetNamespace="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
        schemaLocation="saml-schema-assertion-2.0.xsd"/>
    <import namespace="http://www.w3.org/2000/09/xmldsig#"
        schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-
20020212/xmldsig-core-schema.xsd"/>
    <annotation>
        <documentation>
            Document identity: saml-schema-protocol-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V1.0 (November, 2002):
              Initial Standard Schema.
            V1.1 (September, 2003):
              Updates within the same V1.0 namespace.
            V2.0 (March, 2005):
              New protocol schema based in a SAML V2.0 namespace.
     </documentation>
    </annotation>
…
</schema>
```

8.2.2 请求和应答

接下来的各节定义了 SAML 结构和基本的需求，它们是 SAML 协议中所有请求和应答消息的基础。

8.2.2.1 复杂类型RequestAbstractType

所有的 SAML 请求都是从摘要性质的 **RequestAbstractType** 复杂类型中衍生来的类型。这个类型定义了和所有 SAML 请求有关的通用属性和元素：

注 — 在SAML V2.0中从**RequestAbstractType**去掉了<RespondWith>元素。

— `ID` [必备]

请求的一个标识符。它的类型是 **xs:ID** 并且必须符合第 7.4 节中定义的对标识符唯一性的需求。一个请求中的 ID 属性值必须和相应的应答中的 `InResponseTo` 属性值匹配。

— `Version`[必备]

这个请求的版本。这个建议书中定义的 SAML 版本的标识符为“2.0”。

— `IssueInstant` [必备]

发起请求的时间点。利用 UTC 来编码该值，参见第 7.3 节中的描述。

— `Destination` [可选的]

指示了请求发送到的地址的 URI 引用。这可以防止把请求恶意转发到非目的接收端，有些协议绑定会要求这种保护。如果存在这项内容，那么实际的接收端需要检查标识了接收消息的地点的 URI 引用。如果不存在这项内容，那么丢弃该请求。有些协议绑定可能需要这个属性（参见第 10 节）。

— Consent [可选的]

指示是否(在什么条件下)从发送这个请求的一个责任人处得到了同意或者不同意。有些可能作为 Consent 属性值以及对它们相关描述的值的 URI 引用参见第 8.7.4 节。如果没有提供 Consent 值，那么标识符 urn:oasis:names:tc:SAML:2.0:consent:unspecified 有效。

— <saml:Issuer> [可选的]

标识生成请求消息的实体。

— <ds:Signature> [可选的]

认证请求端以及提供消息完整性的一个 XML 签字，参见下面以及第 8.4 节中的描述 ：

— <Extensions> [可选的]

这个扩展点包括在通信方之间达成一致的可选协议消息扩展元素。使用这种扩展点不需要扩展 Schema，即使提供了扩展 Schema，不严格的确认设置也没有对扩展的认证提出要求。SAML 扩展元素必须符合在非 SAML 定义的命名空间中的命名空间。

根据对特定协议或者协议子集的要求，一个 SAML 请求端可能经常需要认证自己，也可能经常需要消息的完整性。可以通过协议绑定提供的机制来提供认证和消息的完整性（参见第 10 节）。可以给 SAML 请求签字，它提供了对请求端和消息完整性的认证。

如果采用了签字，那么必须出现<ds:Signature>元素，而且 SAML 响应端必须根据 W3C Signature 验证签字的有效性（也就是说，消息没有被篡改）。如果它无效，那么响应端不能依赖请求的内容并且应该应答一个错误消息。如果它有效，那么响应端应该对签字进行评估从而决定签字者的身份以及适应性并且可能继续处理请求或者应答一个错误（因为某些原因，请求可能无效）。

如果包括了 Consent 属性并且值暗示了已经获得了责任人同意的某些形式，那么应该给请求签字。

如果一个 SAML 响应端根据 SAML 句法或者处理规则认为一个请求无效，那么如果它应答的话，它必须返回一个带有值为 urn:oasis:names:tc:SAML:2.0:status:Requester 值的<StatusCode>元素的 SAML 应答消息。在某些情况下，例如在一个可疑的拒绝服务攻击过程中，根本不应答可能是有保证的。

接下来的 Schema 段定义了 **RequestAbstractType** 复杂类型：

```
<complexType name="RequestAbstractType" abstract="true">
       <sequence>
              <element ref="saml:Issuer" minOccurs="0"/>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="samlp:Extensions" minOccurs="0"/>
       </sequence>
       <attribute name="ID" type="ID" use="required"/>
       <attribute name="Version" type="string" use="required"/>
       <attribute name="IssueInstant" type="dateTime" use="required"/>
       <attribute name="Destination" type="anyURI" use="optional"/>
       <attribute name="Consent" type="anyURI" use="optional"/>
</complexType>
<element name="Extensions" type="samlp:ExtensionsType"/>
<complexType name="ExtensionsType">
       <sequence>
              <any namespace="##other" processContents="lax"
maxOccurs="unbounded"/>
       </sequence>
</complexType>
```

8.2.2.2 StatusResponseType复杂类型

所有的 SAML 应答都是从 **StatusResponseType** 复杂类型衍生来的类型。这个类型定义了和所有 SAML 应答有关的通用的属性和元素。

— `ID` [必备]

一个用于应答的标识符。它是类型 **xs:ID**，而且它必须服从第 7.4 节中定义的需求从而保证标识符的唯一性。

— `InResponseTo` [可选的]

如果该项存在，它就是到应答所对应的请求的标识符的一个引用。如果没有针对一个请求产生一个应答，或者如果无法决定一个请求的 `ID` 属性值（例如，请求畸变），那么这个属性一定不能出现。反之，它必须出现而且它的值必须和相应的请求 `ID` 属性相匹配。

— `Version` [必需的]

应答的版本。本建议书中定义的 SAML 的版本的标识符为“2.0”。

— `IssueInstant` [必需的]

发起应答的时间点。通过 UTC 来编码时间值，参见第 7.3 节中的描述。

— `Destination` [可选的]

指示应答目的地的一个 URI 引用。这对于防止将应答恶意转发到非目的接收端的时候有用，某些协议绑定需要这个保护。如果存在这项内容，实际的接收端必须检查标识接收到消息的位置的 URI 引用。如果不存在此项内容，那么必须丢弃应答。某些协议绑定可能需要利用这个属性（参见第 10 节）。

— `Consent` [可选的]

指示在发送这个应答的时候是否（以及在什么条件下）从一个责任人中获得了同意。对于一些可以用做 `Consent` 属性以及和它们相关的描述的 URI 引用参见第 8.7.4 节。如果没有提供 `Consent`，标识符 `urn:oasis:names:tc:SAML:2.0:consent:unspecified`（参见第 8.7.4 节）有效。

— `<saml:Issuer>` [可选的]

标识产生应答消息的实体（需要关于这个元素的更多的信息，参见第 8.1.2.5 节）。

— `<ds:Signature>` [可选的]

认证响应端以及提供消息完整性的一个 XML 签字，参见下面的以及第 8.4 节中的描述：

— `<Extensions>` [可选的]

在通信方之间达成协议的包含可选协议消息扩展元素的扩展点。采用这个扩展点无需扩展 Schema，如果提供了一个扩展 Schema，不严格的确认设置也没有为扩展的有效性提出要求。SAML 扩展元素必须符合非 SAML 定义的命名空间中的命名空间。

— `<Status>` [必需的]

代表相应的请求状态的一个代码。

根据特定协议或者协议子集的需求，一个 SAML 响应端可能经常需要认证它自己，而且经常可能需要消息的完整性。认证和消息完整性可能通过协议绑定提供的机制来提供。可以对 SAML 应答签字，这样就提供了对响应端的认证和消息的完整性。

如果采用了签字，那么必须出现`<ds:Signature>`元素，而且接收应答的 SAML 请求端必须根据 W3C XML Signature 来验证签字的有效性（也就是说，消息没有被篡改）。如果签字无效，那么请求端不能依靠应答的内容而且认为它有错。如果签字有效，那么请求端应该对签字进行评估从而决定签字者的身份和适应性，如果认为合适，那么可能会继续处理应答。

如果包括了 Consent 属性而且值暗示了已经得到了责任人同意的某些形式，那么应该对应答进行签发。

接下来的 Schema 段定义了 **StatusResponseType** 复杂类型：

```
<complexType name="StatusResponseType">
      <sequence>
            <element ref="saml:Issuer" minOccurs="0"/>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="samlp:Extensions" minOccurs="0"/>
            <element ref="samlp:Status"/>
      </sequence>
      <attribute name="ID" type="ID" use="required"/>
      <attribute name="InResponseTo" type="NCName" use="optional"/>
      <attribute name="Version" type="string" use="required"/>
      <attribute name="IssueInstant" type="dateTime" use="required"/>
      <attribute name="Destination" type="anyURI" use="optional"/>
      <attribute name="Consent" type="anyURI" use="optional"/>
</complexType>
```

1） <Status>元素

<Status>元素包含下面的元素：

— <StatusCode> [必需的]

代表在针对相应的请求的应答中执行的行为状态的一个代码。

— <StatusMessage> [可选的]

可能返回给操作者的一条消息。

— <StatusDetail> [可选的]

与请求状态有关的额外信息。

接下来的 Schema 段定义了<Status>元素及其 **StatusType** 复杂类型：

```
<element name="Status" type="samlp:StatusType"/>
<complexType name="StatusType">
    <sequence>
       <element ref="samlp:StatusCode"/>
       <element ref="samlp:StatusMessage" minOccurs="0"/>
       <element ref="samlp:StatusDetail" minOccurs="0"/>
    </sequence>
</complexType>
```

2） <StatusCode>元素

<StatusCode>元素指定了表示相应请求状态的一个代码或者一个嵌套的代码集。<StatusCode>元素具有下面的元素和属性：

— Value [必需的]

状态代码值。这个属性包含了一个URI引用。最高<StatusCode>元素的值必须来自本节给出的最高级别的列表。

— <StatusCode> [可选的]

为一个错误条件提供了更多详细信息的下级状态代码。响应端可以省略下级代码从而防止通过有意提交错误请求而查探额外信息的攻击。

允许的最高级<StatusCode>值如下所示：

```
urn:oasis:names:tc:SAML:2.0:status:Success
```

请求成功。在<StatusMessage>和/或者<StatusDetail>元素中返回额外信息。

```
urn:oasis:names:tc:SAML:2.0:status:Requester
```

因为请求端出现了差错，所有请求不能完成。

```
urn:oasis:names:tc:SAML:2.0:status:Responder
```

因为 SAML 响应端或者 SAML 授权机构出现了差错，所以请求不能完成。

```
urn:oasis:names:tc:SAML:2.0:status:VersionMismatch
```

因为请求信息的版本不对，所以 SAML 响应端无法处理请求。

在本建议书中的不同位置引用了下面的二级状态代码。在 SAML 建议书未来的版本中可能会定义额外的二级状态代码。系统实体可以通过定义适当的 URI 引用来自由地定义更加特殊的状态代码。

```
urn:oasis:names:tc:SAML:2.0:status:AuthnFailed
```

应答提供者无法成功地认证责任人。

```
urn:oasis:names:tc:SAML:2.0:status:InvalidAttrNameOrValue
```

在<saml:Attribute>或者<saml:AttributeValue>元素中遇到了非期望的或者无效的内容。

```
urn:oasis:names:tc:SAML:2.0:status:InvalidNameIDPolicy
```

应答提供者不能或将不支持请求的名字标识符策略。

```
urn:oasis:names:tc:SAML:2.0:status:NoAuthnContext
```

响应端无法满足规定的认证关联需求。

```
urn:oasis:names:tc:SAML:2.0:status:NoAvailableIDP
```

中间媒介利用该代码来指示无法解析出一个<IDPList>中支持的任何一个身份提供者<Loc>元素或者没有支持的身份提供者可用。

```
urn:oasis:names:tc:SAML:2.0:status:NoPassive
```

指示应答提供者无法被动地按照请求动地认证责任人。

```
urn:oasis:names:tc:SAML:2.0:status:NoSupportedIDP
```

中间媒介利用该代码来指示它不支持<IDPList>中的任何一个身份提供者。

```
urn:oasis:names:tc:SAML:2.0:status:PartialLogout
```

一个会话授权机构利用该代码来通知一个会话参与者，它无法把注销消息传递给所有其他的会话参与者。

```
urn:oasis:names:tc:SAML:2.0:status:ProxyCountExceeded
```

指示一个应答提供者无法直接认证责任人而且不被允许继续代理请求。

```
urn:oasis:names:tc:SAML:2.0:status:RequestDenied
```

SAML 响应端或者 SAML 授权机构可以处理请求但是选择不去应答。在考虑请求消息的安全关联的时候或者考虑来自一个特定请求端的请求消息的顺序的时候，需要用到这个状态代码。

```
urn:oasis:names:tc:SAML:2.0:status:RequestUnsupported
```

SAML 响应端或者 SAML 授权机构不支持这个请求。

```
urn:oasis:names:tc:SAML:2.0:status:RequestVersionDeprecated
```

根据请求中指定的协议版本，SAML 响应端无法处理任何请求。

```
urn:oasis:names:tc:SAML:2.0:status:RequestVersionTooHigh
```

SAML 响应端无法处理请求，因为在请求消息中定义的协议版本是响应端所支持最高版本的一个主要的升级版。

```
urn:oasis:names:tc:SAML:2.0:status:RequestVersionTooLow
```

因为请求消息中定义的协议版本太低，所以 SAML 响应端无法处理请求。

```
urn:oasis:names:tc:SAML:2.0:status:ResourceNotRecognized
```

请求消息中提供的资源值无效或者无法识别。

```
urn:oasis:names:tc:SAML:2.0:status:TooManyResponses
```

应答消息中包含多于 SAML 响应端能够返回的元素。

```
urn:oasis:names:tc:SAML:2.0:status:UnknownAttrProfile
```

为不了解特定属性协议子集的一个实体提供了一个来自该协议子集的属性。

```
urn:oasis:names:tc:SAML:2.0:status:UnknownPrincipal
```

应答提供者无法识别请求定义或者暗示的责任人。

```
urn:oasis:names:tc:SAML:2.0:status:UnsupportedBinding
```

SAML 响应端通过请求中定义的协议绑定无法正确完成请求。

接下来 Schema 段定义了<StatusCode>元素及其 **StatusCodeType** 复杂类型：

```
<element name="StatusCode" type="samlp:StatusCodeType"/>
<complexType name="StatusCodeType">
    <sequence>
       <element ref="samlp:StatusCode" minOccurs="0"/>
    </sequence>
    <attribute name="Value" type="anyURI" use="required"/>
</complexType>
```

3） <StatusMessage>元素

<StatusMessage>元素定义了一个可能返回到操作者的消息。

接下来的 Schema 段定义了<StatusMessage>元素：

```
<element name="StatusMessage" type="string"/>
```

4） <StatusDetail>元素

可能通过<StatusDetail>元素来定义涉及请求状态的额外信息。额外信息包含了 0 个或者多个来自任何命名空间的元素，对将要出现的一个模式或者<StatusDetail>内容的模式的有效性没有要求。

接下来的 Schema 段定义了<StatusDetail>元素及其 **StatusDetailType** 复杂类型：

```
<element name="StatusDetail" type="samlp:StatusDetailType"/>
<complexType name="StatusDetailType">
    <sequence>
       <any namespace="##any" processContents="lax" minOccurs="0"
       maxOccurs="unbounded"/>
    </sequence>
</complexType>
```

8.2.3 断言查询和请求协议

本部分定义了利用主体以及断言类型通过引用或者询问断言的方式来请求现存断言的消息以及处理原则。

8.2.3.1 <AssertionIDRequest>元素

如果请求端知道一个或者多个断言的唯一标识符，那么可以利用<AssertionIDRequest>消息元素来请求在一个<Response>消息中返回这些标识符。利用<saml:AssertionIDRef>元素来定义每个需要返回的断言。

接下来的 Schema 段定义了<AssertionIDRequest>元素：

```
<element name="AssertionIDRequest" type="samlp:AssertionIDRequestType"/>
<complexType name="AssertionIDRequestType">
       <complexContent>
              <extension base="samlp:RequestAbstractType">
                     <sequence>
                            <element ref="saml:AssertionIDRef"
maxOccurs="unbounded"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
```

8.2.3.2 询问

接下来的部分定义了 SAML 询问请求消息。

8.2.3.2.1 <SubjectQuery>元素

<SubjectQuery>消息元素是允许定义新的 SAML 询问的一个扩展点，这些新的 SAML 询问指定了一个单一的 SAML 主体。它的 **SubjectQueryAbstractType** 复杂类型是摘要性质的，因此它只能用做一个衍生类型的基础。**SubjectQueryAbstractType** 为 <saml:Subject> 元素（参见第 8.1.4 节中的定义）增加了 **RequestAbstractType**。

接下来的 Schema 段定义了<SubjectQuery>元素及其 **SubjectQueryAbstractType** 复杂类型：

```
<element name="SubjectQuery" type="samlp:SubjectQueryAbstractType"/>
<complexType name="SubjectQueryAbstractType" abstract="true">
     <complexContent>
           <extension base="samlp:RequestAbstractType">
                 <sequence>
                       <element ref="saml:Subject"/>
                 </sequence>
           </extension>
     </complexContent>
</complexType>
```

8.2.3.2.2 <AuthnQuery>元素

通过<AuthnQuery>消息元素询问“对于这个主体哪些包含认证断言的断言有效？”一个成功的<Response>会包括一个或者多个包含认证断言的断言。

对于一个采用请求中提供的信任状的新的认证，不能利用<AuthnQuery>消息作为一个请求。<AuthnQuery>是对一个断言的请求，这是关于出现在指示的主体以及认证授权机构之间的互通过程中的认证行为的一个断言。

这个元素是 **AuthnQueryType** 类型，它通过附加下面的元素和属性扩展了 **SubjectQueryAbstractType**。

— SessionIndex [可选的]

如果出现这项内容，它为可能的应答定义了一个过滤器（filter）。这样一个询问“在所提供的会话信息的关联中，具有哪些用于这个主体的包含认证断言的断言？”

— <RequestedAuthnContext> [可选的]

如果出现这项内容，它为可能的应答定义了一个过滤器。这样一个询问会问“在这个元素中，哪些包含认证断言的断言可以用于满足认证关联需求的主体？”

在应答认证询问的时候，一个 SAML 授权机构返回具有下面认证申明的断言：

- 可能会返回与标识断言的询问的<Subject>元素相匹配的责任人，参见第8.2.3.4节中的描述。
- 如果在询问中出现了SessionIndex属性，在返回的断言集中至少有一个<AuthnStatement>元素必须包含一个和询问中的SessionIndex属性相匹配的SessionIndex属性。可选在应答中是否返回全部匹配断言。
- 如果在询问中出现了<RequestedAuthnContext>元素，那么在返回断言中至少有一个<AuthnStatement>元素必须包含一个满足询问中的元素的<AuthnContext>元素。可选是否在应答中返回所有匹配的断言。

接下来的 Schema 段定义了<AuthnQuery>元素及其 **AuthnQueryType** 复杂类型：

```
<element name="AuthnQuery" type="samlp:AuthnQueryType"/>
<complexType name="AuthnQueryType">
       <complexContent>
              <extension base="samlp:SubjectQueryAbstractType">
                     <sequence>
                            <element ref="samlp:RequestedAuthnContext"
minOccurs="0"/>
                     </sequence>
                     <attribute name="SessionIndex" type="string"
use="optional"/>
              </extension>
       </complexContent>
</complexType>
```

1） <RequestedAuthnContext>元素

<RequestedAuthnContext>元素定义了在应答一个请求或者询问的时候，返回的认证断言的认证关联需求。它的 **RequestedAuthnContextType** 复杂类型定义了下述元素和属性：

— <saml:AuthnContextClassRef>或 <saml:AuthnContextDeclRef> [1个或者多个]

指定了标识认证关联类别或者断言的一个或者多个URI引用。第8.1.7.2.2节定义了这些元素。需要了解更多的关于认证关联类别的内容，参见第12节。

— Comparison [可选的]

定义了用于评估请求的关联类别或者断言的对比方法，它们是“精确”、“最小”、“最大”或者“较好之一”。缺省为“精确”。

可以采用一个类别集也可以采用一个断言引用集。对提供的引用必须按顺序进行评估，第一个元素应该是最优的认证关联类别或者断言。如果根据下面的原则，没有特定的类别或者断言能够满足，那么响应端必须返回一个具有二级 <StatusCode> 为 urn:oasis:names:tc:SAML:2.0:status:NoAuthnContext 的 <Response>消息。

如果将对比设置为“精确”或者省略，那么认证断言中得到的认证关联必须至少精确匹配一个规定的认证关联。

如果将对比设置为“最小”，那么认证断言中得到的认证关联必须至少和给定的认证关联中的一个程度一样（响应端认为）。

如果将对比设置为“较好”，那么认证断言中得到的认证关联必须比规定的认证关联中的一个强壮（响应端认为）。

如果如果将对比设置为“最大”，那么认证断言中得到的认证关联必须尽可能地强壮（响应端认为），但是不能超过至少一个定义的认证关联的强度。

接下来的 Schema 段定义了<RequestedAuthnContext>元素及其 **RequestedAuthnContextType** 复杂类型：

```
<element name="RequestedAuthnContext" type="samlp:RequestedAuthnContextType"/>
<complexType name="RequestedAuthnContextType">
  <choice>
        <element ref="saml:AuthnContextClassRef" maxOccurs="unbounded"/>
        <element ref="saml:AuthnContextDeclRef" maxOccurs="unbounded"/>
  </choice>
  <attribute name="Comparison" type="samlp:AuthnContextComparisonType"
use="optional"/>
</complexType>
<simpleType name="AuthnContextComparisonType">
  <restriction base="string">
        <enumeration value="exact"/>
        <enumeration value="minimum"/>
        <enumeration value="maximum"/>
        <enumeration value="better"/>
  </restriction>
</simpleType>
```

8.2.3.2.3 <AttributeQuery> 元素

利用<AttributeQuery>元素来询问“为这个主体返回请求的属性”。一个成功的应答采用的是包含属性断言的断言的形式，可以扩展的策略允许的范围。这个元素是 **AttributeQueryType** 类型，它通过附加下面的元素扩展了 **SubjectQueryAbstractType**：

— <saml:Attribute> [任何数量]

一个<saml:Attribute>元素，会返回这个属性的值。如果没有规定属性，那么暗示请求了所有策略允许的属性。如果一个给定的<saml:Attribute>元素包含一个或者多个<saml:AttributeValue>元素，那么如果在应答中返回了属性，它一定不能包括任何和询问中定义的值不相等的值。如果不存在特定的协议子集或者属性定义的等同规则，那么定义等同性作为一个相同的值的XML表示法。如需要有关<Saml: Attribute>更详细的信息，参见第8.1.7.3.1节。

一个单一的询问一定不能包含 2 个具有相同的 Name 和 NameFormat 值的<saml:Attribute>元素（也就是说，在一个询问中，一个给定的属性必须只能命名一次）。

在应答一个属性询问的时候，一个 SAML 授权机构需要返回具有如下，属性声明的断言：

- 可能需要返回第8.2.3.4节中给出的用于询问中标识断言的<Subject>元素的规则。
- 如果询问中出现了任何一个<Attribute>元素，它们限制/过滤属性并且按照上面提到的方式有选择地返回的值。
- 可能会通过与应用有关的策略考虑来限制将要返回的属性和值。

可以利用 2 级状态代码 urn:oasis:names:tc:SAML:2.0:status:UnknownAttrProfile 和 urn:oasis:names:tc:SAML:2.0:status:InvalidAttrNameOrValue 通过在一个询问中解释属性或者值来指示问题。

接下来的 Schema 段定义了<AttributeQuery>元素及其 **AttributeQueryType** 复杂类型：

```
<element name="AttributeQuery" type="samlp:AttributeQueryType"/>
<complexType name="AttributeQueryType">
       <complexContent>
              <extension base="samlp:SubjectQueryAbstractType">
                     <sequence>
                            <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
```

8.2.3.2.4 <AuthzDecisionQuery>元素

通过<AuthzDecisionQuery>元素来询问“对于这个主体，是否允许这个资源上的这些行为，给出一个证据？”一个成功的应答应该是以包含授权决定断言的断言的形式出现。

注 — <AuthzDecisionQuery>在SAML V2.0中已经被冻结，没有进一步提高的计划。需要附加功能的用户可以考虑eXtensible Access Control Markup Language(参见X.1142)，它提供了改进的授权决定特征。

这个元素的类型是 **AuthzDecisionQueryType**，它通过增加下面的元素和属性扩展了 **SubjectQueryAbstractType**：

— Resource [必需的]

指示请求授权使用资源的一个 URI 引用。

— <saml:Action> [1 个或者多个]

请求授权的行为。需要更加详细的信息，参见第 8.1.7.4.2 节。

— <saml:Evidence> [可选的]

SAML 授权机构在做授权决定的时候，可能依靠的断言集。关于这个元素的更多信息参见第 8.1.7.4.3 节。

应答一个授权决定询问，一个 SAML 授权机构返回带有如下所示授权决定声明的断言：

- 可能返回用于匹配标识断言的询问的<Subject>元素的规则参见第8.2.3.4节。

接下来的 Schema 段定义了<AuthzDecisionQuery>元素及其 **AuthzDecisionQueryType** 复杂类型：

```
<element name="AuthzDecisionQuery" type="samlp:AuthzDecisionQueryType"/>
<complexType name="AuthzDecisionQueryType">
  <complexContent>
        <extension base="samlp:SubjectQueryAbstractType">
              <sequence>
                    <element ref="saml:Action" maxOccurs="unbounded"/>
                    <element ref="saml:Evidence" minOccurs="0"/>
              </sequence>
              <attribute name="Resource" type="anyURI" use="required"/>
        </extension>
  </complexContent>
</complexType>
```

8.2.3.3 <Response>元素

当一个应答包含 0 个或者多个满足请求的断言，那么采用这个<Response>消息元素。它具有 **ResponseType** 复杂类型，它扩展了 **StatusResponseType** 并且增加了下面的元素：

— <saml:Assertion>或者<saml:EncryptedAssertion> [任意数量]

通过值定义了一个断言，或者可以选择地通过值定义一个加密的断言。需要了解关于这些元素的更详细的信息，参见第 8.1.3.3 节。

接下来的 Schema 段定义了<Response>元素及其 **ResponseType** 复杂类型：

```
<element name="Response" type="samlp:ResponseType"/>
<complexType name="ResponseType">
       <complexContent>
              <extension base="samlp:StatusResponseType">
                    <choice minOccurs="0" maxOccurs="unbounded">
                          <element ref="saml:Assertion"/>
                          <element ref="saml:EncryptedAssertion"/>
                    </choice>
              </extension>
       </complexContent>
</complexType>
```

8.2.3.4 处理原则

在应答一个 SAML 定义的询问消息的时候，SAML 授权机构返回的每个断言必须包含一个<saml:Subject>元素，这个元素必须和询问中的<saml:Subject>元素高度匹配。

当且仅当采用下面的两个条件以后，一个<saml:Subject>元素 S1 才会和 S2 高度匹配。

- 如果S2包含了一个标识符元素（<BaseID>、<NameID>或者<EncryptedID>），那么S1必须包含一个同等的标识符元素，但是S1或者S2中的元素可能加密（或者不加密）。换句话说，S1和S2中标识符的解密格式必须相同。“相同”意味着标识符元素的内容和属性值必须相同。一旦解密，一个加密的标识符必须和最初的标识符相同。
- 如果S2包括一个或者多个<saml:SubjectConfirmation>元素，那么S1必须包括至少一个<saml:SubjectConfirmation>元素，从而可以通过S2中至少一个<saml:SubjectConfirmation>元素描述的方式确认S1。

下面给出了一个什么允许，什么不允许的实例，S1 可能包以一个具有特定格式值的<saml:NameID>，S2 可以包含一个通过加密 S1 的<saml:NameID>元素而得到的<saml:EncryptedID>元素。但是，S1 和 S2 不能包含具有不同格式值和元素内容的一个<saml:NameID>元素，即使认为两个标识符是指同样的责任人。

如果 SAML 授权机构无法提供具有满足一个询问或者一个断言引用所表达的限制条件的任何断言的一个断言，那么<Response>元素一定不能包含一个<Assertion>元素而且必须包含一个具有值为 urn:oasis:names:tc:SAML:2.0:status:Success 的<StatusCode>元素。

必须遵守和底层的请求和应答消息有关的其他所有的处理规则。

8.2.4 认证请求协议

当一个责任人（或者是责任人行为的一个代理）希望获得包含认证断言的一个断言以便于在一个或者多个可信任方上建立一个安全关联，它可以通过认证请求协议来给一个 SAML 授权机构发送一个<AuthnRequest>消息元素并且请求返回一个包含一个或者多个此种断言的<Response>消息。这种断言可能包含任何类型的额外断言，但是至少有一个断言必须包含至少一个认证断言。支持这个协议的一个 SAML 权威机构也被称作一个身份提供者。

除了这个需求之外，返回断言的详细内容和使用的协议子集或者关联有关。而且，没有定义责任人或者代理到身份提供者的认证所采用的确切的方式，但是，采用的认证方式可能会影响到应答的内容。身份提供者涉及的认证有效性问题，或者和认证过程中所涉及的身份提供者和其他实体之间的通信有关的问题不在本协议的讨论范围之内。

下面的描述和处理规则应用了下述的参与者，在采用的特定协议子集中，这些参与者可能是同一个实体：

— 请求端

生成认证请求和应答应该返回到的实体。

— 提交者

给身份提供者提交请求的实体并且在传输消息过程中认证自己，或者根据一个现存的安全关联来建立自己的身份。如果不是请求端，那么提交者在请求端和应答身份提供者之间作为一个中间媒介。

— 被请求的主体

一个实体，针对该实体请求一个或者多个断言。

— 证明实体

被期望能够满足所得到断言的一个<SubjectConfirmation>元素的实体。

— 可信任方

被期望通过断言来实现由正在使用的协议子集或者关联来定义的一个目的，通常是建立一个安全关联。

— 身份提供者

一个实体，提交者把请求提交给该实体，同时提交者从该实体接收到应答。

<AuthnRequest> 元素

为了请求一个身份提供者发布一个具有认证断言的断言，一个提交者需要对身份提供者进行认证（或者依赖一个现存的安全关联）并且发送一个<AuthnRequest>消息，该消息描述了得到的断言应该具有的满足它的目的的特性。这些特性可能是有关断言内容的信息和/或者有关如何把得到的<Response>消息发送给请求端的信息。对提交者的认证处理可能发生在最初传送<AuthnRequest>消息之前、中间或者之后。

请求端可能和请求的提交者不同，例如，请求端是一个试图利用得到的断言来认证或者为被请求的主体授权的可信任方，那么可信任方可以决定是否提供一个业务。

应该通过用来传递消息的协议绑定对<AuthnRequest>消息进行签字或者认证，并提供完整性保护。

这个消息具有 **AuthnRequestType** 复杂类型，它扩展了 **RequestAbstractType** 并且增加了下面的元素和属性，通常这些元素和属性都可选，但是对于某些特定的协议子集可能是必须的。

— <saml:Subject> [可选的]

定义了得到的断言的被请求主体。这可能包括一个或者多个<saml:SubjectConfirmation>元素来指示如何和或通过谁可以确认得到的断言。需要更多的关于这个元素的信息，参见第 8.1.4 节。

如果全部被省略或者没有包括标识符，那么认为消息的提交者是被请求的主体。如果没有包括<saml:SubjectConfirmation>元素，那么认为提交者是唯一需要的证明实体，而且通过正在使用的协议子集和/或者身份提供者的策略来指示使用的方法。

— <NameIDPolicy> [可选的]

为要用来代表被请求主体的名字标识符定义限制。如果省略，可以使用为请求的主体由身份提供者支持的任何类型的标识符，例如可以根据私密性来使用或者限制。

— <saml:Conditions> [可选的]

定义了请求端希望限制有效性以及/或者利用得到断言的 SAML 条件。在响应端认为必要的情况下，它可以修改或者补充这个条件集。把这个元素中的信息作为构成断言的过程的输入，而不是作为利用请求本身的条件。（需要了解关于这个元素的更多的信息，参见第 8.1.5 节。）

— <RequestedAuthnContext> [可选的]

定义需求，如果存在，那么请求端把这些需求加入到认证关联中，应答提供者通过认证关联对提交者进行认证。

— <Scoping> [可选的]

定义了请求端所信任的，用来认证提交者的身份提供者集，同时响应端也定义了和代理<AuthnRequest>消息给后续的身份提供者的有关限制和关联。

— ForceAuthn [可选的]

一个 boolean 值。如果为“真”，身份供应必须直接认证提交者而不是依赖之前的一个安全关联。如果没有提供值，那么缺省值是“假”。但是，如果 ForceAuthn 和 IsPassive 为“真”，那么身份提供者一定不能临时认证提交者，除非对 IsPassive 的限制可以满足。

— IsPassive [可选的]

一个 boolean 值。如果为“真”，身份提供者以及用户代理本身一定不能明显地控制请求端的用户接口而且不能用显而易见的方式和提交者交互。若不提供一个值，缺省为“假”。

— AssertionConsumerServiceIndex [可选的]

非直接标识<Response>消息应该返回到请求端所在的地点。它只是应用在请求端和提交者不同的协议子集中，例如这个建议书中的 Web Browser SSO 协议子集。身份提供者必须通过一个可信任的方式来把属性中的索引值映射到与请求端有关的一个地点。第 9 节提供了一个可能的机制。如果忽略该项，那么身份提供者必须为正在使用的协议子集，把<Response>消息返回到和请求端有关的缺省的地点。如果规定的索引无效，那么身份提供者可能返回一个错误的<Response>或者它可能使用缺省的地点。这个属性和 AssertionConsumerServiceURL 和 ProtocolBinding 属性互斥。

— AssertionConsumerServiceURL [可选的]

通过值定义了<Response>消息必须返回到的地点从而返回给请求端。响应端必须通过某种方式保证定义的值实际上和有关。第 9 节提供了一个可能的机制；给封装的<AuthnRequest>消息签署是另外一个方式。这个属性和 AssertionConsumerServiceIndex 属性是互斥的，而且典型地它是和 ProtocolBinding 属性一起出现。

— ProtocolBinding [可选的]

一个 URI 引用，它定义了在返回<Response>消息的时候，将要使用的协议绑定。需要了解更多的关于此处定义的协议绑定和 URI 引用的信息，参见第 10 节。这个属性和 AssertionConsumerServiceIndex 属性互斥，而且通常和 AssertionConsumerServiceURL 属性一起出现。

— AttributeConsumingServiceIndex [可选的]

非直接定义和请求端有关的信息，请求端描述了请求端希望或者要求身份提供者在<Response>消息中提供的 SAML 属性。身份提供者必须通过一个可信任的方式把属性中的索引值映射为和请求端有关的信息。第 9 节提供了一个可能的机制。身份提供者可能通过这个信息在返回的断言中携带一个或者多个<saml:AttributeStatement>元素。

— ProviderName [可选的]

为请求端定义了人类可读的名字，提交者的用户代理或者身份提供者会用到这个名字。

第 8.2.4.4 节中给出了关于这个消息的通用处理规则。

接下来的 Schema 段定义了<AuthnRequest>元素及其 **AuthnRequestType** 复杂类型：

```
<element name="AuthnRequest" type="samlp:AuthnRequestType"/>
<complexType name="AuthnRequestType">
      <complexContent>
            <extension base="samlp:RequestAbstractType">
                  <sequence>
                        <element ref="saml:Subject" minOccurs="0"/>
                        <element ref="samlp:NameIDPolicy" minOccurs="0"/>
                        <element ref="saml:Conditions" minOccurs="0"/>
                        <element ref="samlp:RequestedAuthnContext"
minOccurs="0"/>
                        <element ref="samlp:Scoping" minOccurs="0"/>
                  </sequence>
                  <attribute name="ForceAuthn" type="boolean"
use="optional"/>
                  <attribute name="IsPassive" type="boolean"
use="optional"/>
                  <attribute name="ProtocolBinding" type="anyURI"
use="optional"/>
                  <attribute name="AssertionConsumerServiceIndex"
type="unsignedShort" use="optional"/>
                  <attribute name="AssertionConsumerServiceURL"
type="anyURI" use="optional"/>
                  <attribute name="AttributeConsumingServiceIndex"
type="unsignedShort" use="optional"/>
                  <attribute name="ProviderName" type="string"
use="optional"/>
            </extension>
      </complexContent>
</complexType>
```

8.2.4.1 <NameIDPolicy>元素

<NameIDPolicy>元素在来自<AuthnRequest>的断言主体中定制了名字标识符。它的 **NameIDPolicyType** 复杂类型定义了下面的属性：

— Format [可选的]

定义了对应在这个或者另外的建议书中定义的一个名字标识符格式的 URI 引用（参见第 8.7.3 节中给出的实例）。特地为 urn:oasis:names:tc:SAML:2.0:nameid-format:encrypted 定义了用在本属性中的附加值，用来指示对得到的标识符进行加密的一个请求。

— SPNameQualifier [可选的]

有选择地定义了在一个服务提供商的命名空间中（或生成的），或者是在服务提供商的关联组的命名空间中将要返回的断言主体的标识符。参见这个建议书中关于 urn:oasis:names:tc:SAML:2.0:nameid-format:persistent 的定义的实例。

— AllowCreate [可选的]

一个 boolean 值，在完成请求的过程中，它用来指示身份提供者是否被允许生成一个新的标识符来表示责任人。缺省为“假”，如果为“假”，如果已经建立了一个可接受的用于责任人的标识符，那么请求端限制身份提供者只能给它发布一个断言。这无法阻止身份提供者生成在这个特定请求的关联之外的标识符（例如，预先为大量的责任人生成标识符）。

注 1（资料性的）— PE14（参见OASIS PE：2006）澄清了上面的定义，如下所示：

一个boolean值，它用来指示在完成请求的过程中，请求端是否同意身份提供者生成一个新的标识符或者将把一个现存的代表责任人的标识符和一个可信任方关联起来。如果没有此项或者省略整个元素，那么缺省为“假”。

注 2（资料性的）— PE14（参见OASIS PE:2006）建议把下面的文字加入到下面的段落中：

有些部署可能会利用AllowCreate属性来影响由身份提供者维护的状态的生成，从而适合一个特定的可信任方使用一个名字标识符（或者其他任何稳定的、唯一的标识属性），它的目的是用在动态标识符的或者属性的生成、同意的跟踪、名字标识符管理协议的后续使用或者用于其他相关的目的。

如果为“假”，只有在这种状态已经建立起来或者认为身份提供者使用一个标识符不可行，那么请求端会试图限制身份提供者发布一个断言。因此，这无法阻止身份提供者认为这些信息不在这个特定请求的关联之内（例如，事先为大量的责任人建立它）。

值为“真”，那么就允许身份提供者采用任何相关的行为来完成请求，遵循任何由请求或者策略提出的限制条件（例如，IsPassive属性）。

通常，请求端不能假定身份提供者的特定行为考虑了初始标识符的生成或者关联，这些是实施或者部署需要考虑的细节。缺少管理这个属性的使用的特定的协议子集，可以用它来身份提供者提供一个暗示，暗示请求端存储标识符的目的，或者将它链接到一个本地值。

没有特别使用这个属性的请求端通常应该把它置“真”，从而最大化互通性。在和请求结合使用的时候，一定不能使用AllowCreate属性而且应该忽略该属性，或者在发布带有名字标识符的断言的时候，如果名字标识符的Fromat为urn:oasis:names:tc:SAML:2.0:nameid- format:transient的时候，也不能使用该属性（它们排除任何此类的状态）。

在采用这个元素的时候，如果身份提供者无法理解或者接受它的内容，那么必须返回一个带有<Status>差错的<Response>消息元素，而且可能包含一个值为urn:oasis:names:tc:SAML:2.0:status: InvalidNameIDPolicy的二级<StatusCode>。

如果忽略了 Format 值或者将 Format 值设为 urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified，那么身份提供者可以自由地返回任何类型的标识符主体，给任何提供该元素的内容或身份提供者或者责任人的策略，但是需要符合这个元素的内容或者身份提供者或者责任人的策略提出来的附加的限制。

特定的 Fromat 值 urn:oasis:names:tc:SAML:2.0:nameid-format:encrypted 指示得到的断言必须包含<EncryptedID>元素而不是包含纯文本。对于请求的主体来讲，下面的名字标识符的解密格式可以是任何一个身份提供者所支持的类型。

注 3（资料性的）— PE6（参见OASIS PE:2006）建议把下面的文本加入到上面段落的末尾。

若它请求加密，服务提供商不可能特别请求返回一个特定类型的需要加密的标识符。可以采用第9节中的<md:NameIDFormat>元数据元素或者其他带外方式来决定为何种类型的标识符加密并且返回。

注 4（资料性的）— PE15（参见OASIS PE:2006）建议加入下面的段落：

如果采用第 8.7.3.7 节中定义的一个格式而不是采用 urn:oasis:names:TC:SAML:2.0:nameid-format:unspecified 或者 urn:oasis:names:TC:SAML:2.0:nameid-format:encrypted，那么如果身份提供者返回任何的断言：

- 任何一个 <Assertion> 的 <Subject> 内的 <NameID> 的 Format 值必须和<NameIDPolicy>中提供的Format值相同；而且
- 如果没有忽略<NameIDPolicy>中的SPNameQualifier，那么任何<Assertion>的<Subject>中的<NameID>的SPNameQualifier值必须和<NameIDPolicy>所提供的SPNameQualifier值相同。

不考虑<NameIDPolicy>中的格式，如果身份提供者要求采用一个加密的标识符的策略有效（可能是服务提供商所特有的），那么身份提供者可以在得到的断言主体中返回一个<EncryptedID>。

如果责任人不具有标识符，而且如果请求端希望允许身份提供者为责任人建立一个新的标识符，它必须包括这个元素，同时把 AllowCreate 属性置为“真”。否则，只有身份提供者已经为之建立一个请求端可用的标识符的责任人可以成功地得到认证。这主要是在和 urn:oasis:names:tc:SAML:2.0:nameid-format:persistent Format 值相结合的时候有用（参见第 12 节）。

注 5（资料性的）— PE14（参见 OASIS PE:2006）建议忽略上面的段落。

接下来的 Schema 段定义了<NameIDPolicy>元素及其 **NameIDPolicyType** 复杂类型：

```
<element name="NameIDPolicy" type="samlp:NameIDPolicyType"/>
<complexType name="NameIDPolicyType">
      <attribute name="Format" type="anyURI" use="optional"/>
      <attribute name="SPNameQualifier" type="string" use="optional"/>
      <attribute name="AllowCreate" type="boolean" use="optional"/>
</complexType>
```

8.2.4.2 <Scoping>元素

<Scoping>元素定义了请求端信任的身份提供者来对提交者进行认证，同时还根据响应端定义了和代理<AuthnRequest>消息到后续身份提供者有关的限制和关联。它的 **ScopingType** 复杂类型定义了下面的元素和属性：

— ProxyCount [可选的]

定义了在接收到这个<AuthnRequest>的身份提供者和最终验证这个责任人的身份提供者之间允许的间接代理的数量。0 是指没有代理同时利用省略这个属性来标识没有限制。

— <IDPList> [可选的]

请求端认为应答请求可以接受的一个关于身份提供者的建议列表和相关信息。

— <RequesterID> [0 个或者多个]

标识请求端正在采取动作的请求实体集。当出现一系列代理的时候，用来交流一系列的请求端，参见第 8.2.4.5 节中的描述。对实体标识符的描述参见第 8.7.3.6 节。

在定义了一个激活的中间媒介的协议子集过程中，中间媒介可能会检查该列表，如果它无法联系或者不支持任何一个指定的身份提供者，它会返回一个带有差错<Status>和一个值为 urn:oasis:names:tc:SAML:2.0:status:NoAvailableIDP 或者 urn:oasis:names:tc:SAML:2.0:status:NoSupportedIDP 的二级<StatusCode>的<Response>消息。

接下来的 Schema 段定义了<Scoping>元素及其 **ScopingType** 复杂类型：

```
<element name="Scoping" type="samlp:ScopingType"/>
<complexType name="ScopingType">
      <sequence>
            <element ref="samlp:IDPList" minOccurs="0"/>
            <element ref="samlp:RequesterID" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
      <attribute name="ProxyCount" type="nonNegativeInteger"
use="optional"/>
</complexType>
<element name="RequesterID" type="anyURI"/>
```

8.2.4.3 <IDPList>元素

<IDPList>元素定义了请求端信任的身份提供者来对提交者进行验证。它的 **IDPListType** 复杂类型定义了下面的元素：

— <IDPEntry> [1 个或者多个]

关于一个身份提供者的信息。

— <GetComplete> [可选的]

如果<IDPList>不完全，通过这个元素来定义一个 URI 引用，通过该引用来从新获得完整的列表。重新获得与 URI 有关的资源必须导致一个根元素为<IDPList>的 XML 实例自己没有包含一个<GetComplete>元素。

接下来的 Schema 段定义了<IDPList>元素及其 **IDPListType** 复杂类型：

```
<element name="IDPList" type="samlp:IDPListType"/>
<complexType name="IDPListType">
      <sequence>
            <element ref="samlp:IDPEntry" maxOccurs="unbounded"/>
            <element ref="samlp:GetComplete" minOccurs="0"/>
      </sequence>
</complexType>
<element name="GetComplete" type="anyURI"/>
```

<IDPEntry>元素定义了一个请求端信任的单一的身份提供者来对提交者进行认证。它的 **IDPEntryType** 复杂类型定义了下面的属性：

— ProviderID [必备]

身份提供者的唯一的标识符。对于这样的标识符的描述参见第 8.7.3.6 节。

— Name [可选的]

身份提供者的人类可读的名字。

— Loc [可选的]

一个 URI 引用，它标识了支持认证请求协议的具有特定协议子集的端点的位置。采用的协议子集必须可以理解将要使用的绑定。

接下来的 Schema 段定义了<IDPEntry>元素及其 **IDPEntryType** 复杂类型：

```
<element name="IDPEntry" type="samlp:IDPEntryType"/>
<complexType name="IDPEntryType">
      <attribute name="ProviderID" type="anyURI" use="required"/>
      <attribute name="Name" type="string" use="optional"/>
      <attribute name="Loc" type="anyURI" use="optional"/>
</complexType>
```

8.2.4.4 处理规则

<AuthnRequest>和<Response>互换支持一系列的使用方案，因此典型地它被定制协议子集以用在特定的环境中，在这个环境中，这种选择性会受到限制同时要求或者禁止特定种类的输入和输出。把接下来的处理规则作为不变的行为应用在这个协议互换的任何一个协议子集中。和下层请求以及应答消息有关的其他所有处理规则也必须遵守。

响应端最终必须为<AuthnRequest>应答一个<Response>消息，该消息包含一个或者多个满足请求定义的规范的断言，或者该消息包含一个描述出现的差错的<Status>。响应端可以管理额外的消息交换，但是需要提交者发起或者完成认证过程，同时需要服从协议绑定和认证机制的特点。下面的各节会介绍到，这包括通过发布提交者自己的<AuthnRequest>消息，把提交者指引到另外一个身份提供者来代理请求，于是可以通过得到的断言来认证提交者直到起始的响应端，实际上是把 SAML 作为认证机制。

如果响应端无法认证提交者或者无法识别请求的主体，或者无法通过在身份提供者处起作用的策略来提供一个断言（例如，目的主体禁止身份提供者向可信任方提供断言），那么它必须返回一个带有差错<Status>的<Response>，而且也可能返回一个值为 urn:oasis:names:tc:SAML:2.0:status:AuthnFailed 或者 urn:oasis:names:tc:SAML:2.0:status:UnknownPrincipal 的二级<StatusCode>。

如果在请求中出现了<saml:Subject>元素，那么得到断言的<saml:Subject>必须严格匹配请求<saml:Subject>，参见第 8.2.3.4 节中的描述，一个特例就是由<NameIDPolicy>定义的标识符可能采用了不同的格式。在这种情况下，标识符的物理内容可能不同，但是它必须代表相同的责任人。

所有在<AuthnRequest>中特殊定义的内容都是可选的，但是有些可能是特定的协议子集所必须的。如果根本不存在任何特定的内容，那么暗指执行下面的行为：

- 返回的断言必须包含一个代表提交者的<saml:Subject>元素。由身份提供者来决定标识符的类型和格式。至少一个断言中的一个断言必须是<saml:AuthnStatement>，它用来描述由响应端或者和它有关的认证业务执行的认证。

- 请求提交者应该尽可能地作为唯一的能够满足断言的<saml:SubjectConfirmation>的证明实体。在较弱的确认方式下，可以通过特定的映射或者其他机制来帮助满足这个需求。
- 得到的断言必须包含一个<saml:AudienceRestriction>元素，把请求端作为一个可接受的可信任方。可以包括其他被身份提供者认为适合的用户。

8.2.4.5 代理

如果接收到一个<AuthnRequest>的身份提供者还没有认证提交者或者无法直接认证提交者，但是相信提交者已经得到了另外的身份提供者的认证或者是一个非 SMAL 等价体，它可能根据自己的行为向其他的身份提供者通过发布一个新的<AuthnRequest>来应答请求，最初的身份提供者的术语是代理身份提供者。

在成功地给代理提供者返回一个<Response>(或者非SAML等价体)后，可以采用封装的断言或者非SAML等价体来认证提交者，因此代理提供者可以发布一个它自己的断言来应答最初的<AuthnRequest>，从而完成整个的消息交换。代理和认证身份提供者都可能在消息中包含对代理行为和他们发布的断言的限制条件，参见前面以及后面各节的描述。

请求端可以通过包含一个<Scoping>元素来影响代理行为，此时提供者设置一个期望的 ProxyCount 值和/或来指示一个优选提供者列表，它可以由通过包含一个有顺序的优选身份提供者的<IDPList>来进行代理。

一个身份提供者可以利用它发布的断言中的<ProxyRestriction>元素通过代理身份提供者来控制它的断言的二次使用。

如果忽略了<ProxyCount>属性或者它的值大于 0，那么一个身份提供者可以代理一个<AuthnRequest>。它是否选择代理一个<AuthnRequest>是本地策略的事。如果提供了<IDPList>，那么一个身份提供者可以选择代理<IDPList>中指定的一个提供者，但是并不是必须这样做。

当<ProxyCount>置为 0 的时候，一个身份提供者一定不能代理一个请求。除非身份提供者能够直接认证提交者，那么它必须返回一个包含了一个值为 urn:oasis:names:tc:SAML:2.0:status:ProxyCountExceeded 的二级<StatusCode>的差错<Status>。

如果它选择代理一个 SAML 身份提供者，在生成新的<AuthnRequest>的时候，代理身份提供者必须包括等价物或者对包含在起始请求（例如认证关联策略）中的所有信息来讲更加严格的格式。但是，代理提供者可以自由地定义它所采用的<NameIDPolicy>从而最大化成功应答的机会。

如果认证身份提供者不是一个 SAML 身份提供者，那么代理提供者必须有办法来保证认证提供者能够承认管理用户代理交互（例如，<IsPassive>）的元素。

新的<AuthnRequest>必须包含一个<ProxyCount>属性，它的值最多比起始的值少 1。如果起始的请求没有包含<ProxyCount>属性，那么新的请求应该包含一个<ProxyCount>属性。

如果在最初的请求中定义了一个<IDPList>，那么新的请求必须也包含一个<IDPList>。代理身份提供者可以在<IDPList>的末尾增加额外的身份提供者，但是一定不能从列表中去掉任何一个身份提供者。

根据本节中给定的规则和采用的协议子集，通过正常的方式处理认证请求和应答。一旦提交者得到了代理身份提供者的认证，（在投递<Response>的 SAML 的情况下）接下来的步骤如下：

- 代理身份提供者根据它自己的行为，通过从起始的断言或者非SAML等价体中复制相关的信息来准备一个新的断言。
- 新的断言的<saml:Subject>必须包含一个满足最初的请求端的偏好的标识符，它的<NameIDPolicy>元素对此进行了定义。

- 新的断言中的<saml:AuthnStatement>必须包括一个<saml:AuthnContext>元素，该元素包含一个指向身份提供者的<saml:AuthenticatingAuthority>元素，对于这个元素来讲，代理身份提供者被看做是提交者。如果最初的断言包含<saml:AuthnContext>信息，这个信息包含一个或者多个<saml:AuthenticatingAuthority>元素，那么新的断言应该包含这些元素，新的元素放在这些元素的后面。
- 如果认证身份提供者不是一个SAML提供者，那么代理身份提供者必须为认证提供者生成一个唯一的标识符值。在不同的请求过程中，这个值应该一致。这个值一定不能和由其他的SAML提供者使用或者生成的值相冲突。
- 根据代理身份提供者的策略，任何其他的<saml:AuthnContext>信息都可能被复制，翻译或者省略，假定需要满足由请求端提出来的起始需求。

如果，在未来的时间里，要求身份提供者为第二个请求端来验证同样的提交者，而且这个请求和最初的请求具有相同的或者更低的限制（由代理身份提供者来决定），那么身份提供者可能不会为认证身份提供者生成一个新的<AuthnRequest>同时立即发布一个断言（假设最初接收到的断言或者非 SAML 等价体继续有效）。

8.2.5 凭证解析协议

凭证解析协议提供了一种机制，通过该机制，可以通过引用而不是值在 SAML 绑定中传递 SAML 协议消息。利用这种特殊的协议，通过引用可以得到请求和应答。一个消息发送端不用把一个消息映射到一个传输协议，而可以通过绑定发送一小片被称作凭证的数据即可。凭证可以采用不同的格式，但是必须支持的一种方式是，接收端通过该方式判断是谁发送了这个凭证。如果接收端愿意，他可以把这个协议和不同的（通常是同步的）SAML 绑定协议结合起来把凭证解析为起始的协议消息。

这个机制的一个最常见的使用情况就是和因为大小限制原因无法轻松地携带一个消息的映射一起使用，或者是通过一个安全隧道在 SAML 请求端和响应端之间交流一个消息，从而避免使用签字。

根据通过引用来传递的底层消息的特点，凭证解析协议可能需要来自解析凭证的协议绑定的一些安全机制例如互认证、完整性保护和机密性等。在所有的情况下，凭证必须采用一个单次使用语义，一旦它被成功解析，任何团体都不能在使用它。

无需考虑得到的协议消息，必须正确地对待解析一个凭证的结果，好像已经把通过这种方式得到的消息发送到最初凭证所在的位置。

8.2.5.1 <ArtifactResolve> 元素

<ArtifactResolve>消息用来请求通过定义一个代表 SAML 协议消息的凭证，在一个<ArtifactResponse>消息中返回一个 SAML 协议消息。由采用的特定的协议绑定来管理凭证的最初传输。

利用传递消息的协议绑定来给<ArtifactResolve>消息签字或者进行认证和提供完整性保护。

这个消息具有复杂类型 **ArtifactResolveType**，它扩展了 **RequestAbstractType** 而且增加了下面的元素。

— <Artifact> [必需的]

请求端接收到的凭证值，现在希望翻译成它代表的协议消息。

接下来的 Schema 段定义了<ArtifactResolve>元素及其 **ArtifactResolveType** 复杂类型：

```
<element name="ArtifactResolve" type="samlp:ArtifactResolveType"/>
<complexType name="ArtifactResolveType">
      <complexContent>
            <extension base="samlp:RequestAbstractType">
                  <sequence>
                        <element ref="samlp:Artifact"/>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
<element name="Artifact" type="string"/>
```

8.2.5.2 <ArtifactResponse>元素

一个<ArtifactResolve>消息的接收端必须应答一个<ArtifactResponse>消息元素。这个元素的复杂类型是 **ArtifactResponseType**，它通过一个对应将要返回的 SAML 协议消息的可选的通配符元素扩展了 **StatusResponseType**。该通配符消息元素可以是一个请求端或一个响应端。

通过传递消息的协议绑定来为<ArtifactResponse>签字或者进行认证并且提供完整性保护。

接下来的 Schema 段定义了<ArtifactResponse>元素及其 **ArtifactResponseType** 复杂类型：

```
<element name="ArtifactResponse" type="samlp:ArtifactResponseType"/>
<complexType name="ArtifactResponseType">
      <complexContent>
            <extension base="samlp:StatusResponseType">
                  <sequence>
                        <any namespace="##any" processContents="lax"
minOccurs="0"/>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
```

8.2.5.3 处理规则

如果响应端认为凭证有效，那么它在一个<ArtifactResponse>消息元素中应答一个相关的协议消息。否则，它应答一个没有内嵌消息的<ArtifactResponse>元素。在这两种情况下，<Status>元素必须包括一个编码值为 urn:oasis:names:tc:SAML:2.0:status:Success 的<StatusCode>元素。在本部分接下来的部分，一个没有内嵌消息的应答消息被称为空应答。

响应端通过保证来自任何请求端的具有同样凭证的请求都会得到一个如上所述的空的应答，从而必须给凭证强加一个一次使用的特性。

一些 SAML 协议消息，最常见的是一些协议子集中的<AuthnRequest>消息，可能被任何接收到它的团体所使用并且适当地应答。但是，在其他大部分情况下，一个消息可能用于一个特定的实体。在这种情况下，在发布时，发布的凭证必须和凭证所代表的这个消息的目的接收端想关联。如果凭证发布者从一个请求端接收到了一个<ArtifactResolve>消息，而该请求端无法像起始的目的接收端那样验证自己，那么凭证发布者必须返回一个空应答。

在使用一个凭证的时候，凭证发布者应该强制使用一个最短参与时间限制，那么存在一个凭证接收端可以获得凭证的可接受的时间窗口并且通过一个<ArtifactResolve>消息将这个时间窗口返回给发布者。

<ArtifactResponse>消息的 InResponseTo 属性必须包含相应的<ArtifactResolve>消息的 ID 属性的值，但是内嵌的协议消息需要包含它自己的消息标识符，而且如果存在内嵌的应答，它可能包含一个不同的 InResponseTo 值，该值对应起始的内嵌消息正在应答的请求消息。

必须遵守所有其他的和下层请求和应答消息有关的处理规则。

8.2.6 名字标识符管理协议

在为一个责任人建立了一个名字标识符以后，在引用一个原则的时候，一个身份提供者希望改变它将要使用的标识符的值以及/或者格式，或者指示不能再利用一个名字标识符来引用一个责任人，通过给服务提供商发送一个<ManageNameIDRequest>消息，来通知他们所发生的变化。

注 1（资料性的）— PE12（参见OASIS PE:2006）通过改写上述段落来表示了它的意图：

在为一个责任人建立了一个名字标识符以后，一个身份提供者在引用一个责任人的时候，希望改变它将要使用的标识符的值或者指示不能再利用一个一个名字标识符来应用该责任人，通过给服务提供商发送一个<ManageNameIDRequest>消息来通知服务提供商所发生的变化。

当利用底层的名字标识符来和服务提供商进行通信的时候，服务提供商还使用这个消息来注册或者改变将要包含的 SPProvidedID 值，或者终结在服务提供商与身份提供者之间使用的一个名字标识符。

典型地，这个协议不会利用“暂时”的名字标识符，因为可能无法长时间地管理它们的值。

注 2（资料性的）— PE14（参见OASIS PE:2006）澄清了上述的问题，如下所示：

这个协议一定不能和 urn:oasis:names:tc:SAML:2.0:nameidformat:transient <NameID> Format 结合使用。

8.2.6.1 <ManageNameIDRequest>元素

一个提供者发送一个<ManageNameIDRequest>消息来通知一个发生变化的名字标识符的接收端或者指示终结一个名字标识符的使用。

应该通过用来传递消息的协议绑定来对<ManageNameIDRequest>消息进行签字或者进行认证和提供完整性保护。

这个消息具有复杂类型 **ManageNameIDRequestType**，它扩展了 **RequestAbstractType** 并且增加了下面的元素：

— <saml:NameID>或者<saml:EncryptedID> [必需的]

定义了责任人的名字标识符和相关的描述行数据（通过纯文本或者加密的格式），在这个请求之前（希望了解关于这些元素的更多的信息，参见第 8.1.2 节）由身份和服务提供商识别。

— <NewID>或者<NewEncryptedID>或者<Terminate> [必需的]

在和考虑这个责任人的正在请求的提供者进行通信的时候，采用的新的标识符值（采用纯文本或者加密的格式），或者指示终止了对旧的标识符的使用。在前一种情况中，如果请求端是服务提供商，那么新的标识符必须出现在 SPProvidedID 属性的接下来的<NameID>元素中。如果请求端是身份提供者，那么新值会作为元素的内容出现在接下来的<NameID>元素中。

注（资料性的）— PE12（参见OASIS PE:2006）建议把下面的内容附加到上面的段落。

在任何一种情况下，如果采用了<NewEncryptedID>，那么它的加密内容就是<NewID>元素，该元素只包含该标识符的新值（一旦建立好了以后，格式和限定词不能改变）。

接下来的 Schema 段定义了<ManageNameIDRequest>元素及其 **ManageNameIDRequestType** 复杂类型：

```
<element name="ManageNameIDRequest" type="samlp:ManageNameIDRequestType"/>
<complexType name="ManageNameIDRequestType">
      <complexContent>
            <extension base="samlp:RequestAbstractType">
                  <sequence>
                        <choice>
                              <element ref="saml:NameID"/>
                              <element ref="saml:EncryptedID"/>
                        </choice>
                        <choice>
                              <element ref="samlp:NewID"/>
                              <element ref="samlp:NewEncryptedID"/>
                              <element ref="samlp:Terminate"/>
                        </choice>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
```

```
<element name="NewID" type="string"/>
<element name="NewEncryptedID" type="saml:EncryptedElementType"/>
<element name="Terminate" type="samlp:TerminateType"/>
<complexType name="TerminateType"/>
```

8.2.6.2 <ManageNameIDResponse>元素

一个<ManageNameIDRequest>消息的接收端必须应答一个<ManageNameIDResponse>消息，它是**StatusResponseType** 类型，没有额外的内容。

应该利用传递消息的协议绑定来对<ManageNameIDResponse>消息签字或者进行认证以及完整性保护。

接下来的 Schema 段定义了<ManageNameIDResponse>元素：

```
<element name="ManageNameIDResponse" type="samlp:StatusResponseType"/>
```

8.2.6.3 处理规则

如果请求包含了一个接收端无法识别的<saml:NameID>（或者加密版本），那么应答提供者必须应答一个差错<Status>而且可能应答一个值为 urn:oasis:names:tc:SAML:2.0:status:UnknownPrincipal 的二级<StatusCode>。

注 1（资料性的）— PE14（参见OASIS PE:2006）进一步澄清了下面的段落。需要更详细的内容参见附录八。

如果<Terminate>元素包含在请求中，那么请求提供者就暗示（在一个服务提供商情况下）它不再接受来自身份提供者的断言或者（在身份提供者的情况下）它不再向服务提供商发布关于责任人的断言。接收提供者可以执行任何的维护，它知道由名字标识符所代表的关系已经终止。它可以选择让一个关系已经被终止的责任人的工作会话无效。

注 2（资料性的）— PE8（参见OASIS PE:2006）建议利用下面的内容替代本段中的最后一句话：

通常，它不应该使一个关系已经被终止的责任人的任何一个工作会话失效。如果接收提供者是一个身份提供者，那么它不应该让任何和其他服务提供商之间已经建立的责任人的工作会话失效。如果请求提供者愿意（例如，可以通过一个希望终止所有用户行为的管理者来发起名字标识符终止动作），它可以在通过发送一个<ManageNameIDRequest>消息来发起一个名字标识符终止之前发送一个<LogoutRequest>消息。在发送了<ManageNameIDRequest>消息以后，请求提供者一定不能发送一个<LogoutRequest>消息。

如果服务提供商请求通过包含一个<NewID> (或者<NewEncryptedID>)元素来改变用于责任人的标识符，当后续的和涉及这个责任人的服务提供商之间的通信时候，身份提供者必须包含元素的内容 SPProvidedID 的内容。

如果身份提供者请求通过包含一个<NewID> (或者<NewEncryptedID>)元素来改变用于责任人的标识符，当后续的和身份提供者之间的通信涉及了这个责任人的时候，服务提供商必须把元素的内容作为责任人<saml:NameID>的元素内容。

最初的和新的标识符可以不加密，也可以加密一个或者两个都加密（采用<EncryptedID> 和<NewEncryptedID>元素）。

在任何一种情况下，请求中<saml:NameID>的内容及其相关的 SPProvidedID 属性必须包含提供者之间为责任人建立的最新的名字标识符信息。

在一个标识符的 Format 为 urn:oasis:names:tc:SAML:2.0:nameid-format:persistent 的情况下，NameQualifier 属性必须包含生成标识符的身份提供者的唯一的标识符。如果在身份提供者和一个联盟组之间建立了标识符，服务提供商是联盟的一个成员，那么 SPNameQualifier 属性必须包含联盟组的唯一的标识符。否则，它必须包含服务提供商的唯一的标识符。如果这些属性和所包含的信息的<Issuer>元素的值相匹配，那么可以忽略这些属性，但是因为可能出现冲突，并不建议采用这种方式。

对这些标识符的改变可能要花费大量的时间来在请求端和响应端所处的系统内传播。在成功完成一个名字标识符的更改以后，执行可能会希望允许每一方在一个时间内接收任何一个标识符。如果不采取这种方式，可能会导致责任人无法访问资源。

必须遵守其他的和底层请求和应答消息有关的处理规则。

8.2.7 单点注销协议

单点注销协议提供了一个消息交换协议，通过这个协议，由一个特定会话授权机构提供的所有的会话基本是同步终结。当一个责任人在一个会话参与者处注销的时候或者当责任人直接在会话授权机构注销的时候采用单点注销协议。这个协议也可以用在根据超时而注销一个责任人的情况下。可以通过 Reason 属性来指示出现注销事件的原因。

根据包含由会话授权机构提供的认证断言的断言，责任人可能已经和会话授权机构和独立的会话参与者之间建立了得到认证的会话。

当责任人在一个会话参与者处调用单点注销程序的时候，会话参与者必须向会话授权机构发送一个<LogoutRequest>消息，这个授权机构提供了包含和在会话参与者处会话有关的认证断言的断言。

不管是责任人在会话授权机构处调用一个注销程序，还是会话参与者给定义了责任人的会话授权机构发送一个注销请求，会话授权机构应该给每个会话参与者发送一个<LogoutRequest>消息，对于每个会话参与者来讲，它发送的断言包含了它和责任人的当前会话下的认证断言，会话参与者的一个特例就是它向会话权威机构发送一个<LogoutRequest>消息。它应该尽可能地多联系这些参与者，就像它尽可能地利用这个协议一样，终结它自己和责任人之间的会话，并且最终给，若有，任何一个请求会话参与者返回一个<LogoutResponse>消息。

8.2.7.1 <LogoutRequest>元素

一个会话参与者或者会话授权机构通过发送一个<LogoutRequest>消息来指出一个会话已经终结。

通过用来传递消息的协议绑定对<LogoutRequest>消息进行签字或者进行认证并提供完整性保护。

这个消息具有 **LogoutRequestType** 复杂类型，它扩展了 **RequestAbstractType** 并且增加了下面的元素和属性：

— NotOnOrAfter [可选的]

请求过期的时间，这个时间以后，接收端可能会丢弃该消息。通过 UTC 来编码时间值，参见第 7.3 节中的描述。

— Reason [可选的]

利用一个 URI 引用的格式来指出注销的原因。

注 1（资料性的）— PE10（参见OASIS PE:2006）建议利用下面的内容代替上面的文本：
Schema中把Reason属性定义为一个字符串。这个规范通过要求Reason属性必须采用URI引用格式而进一步限制了Schema。

— <saml:BaseID>或者<saml:NameID>或者<saml:EncryptedID> [必需的]

标识符和相关的属性（纯文本格式或者加密格式），它们定义了在这个请求之前，当前被身份和服务提供商所识别的责任人（需要关于这个元素的更多的信息，参见第 8.1.2 节）。

— <SessionIndex> [可选的]

在消息接收端处指出这个会话的标识符。

注 2（资料性的）— PE38（参见OASIS PE:2006）澄清了上面的文本，如下所示：
由<saml:BaseID>、<saml:NameID>或者<saml:EncryptedID>标识的责任人和会话授权机构之间的会话索引。它必须和断言中<saml:AuthnStatement>中的SessionIndex属性（如果存在）关联起来，从而建立正在终结的会话。

接下来的片段定义了<LogoutRequest>元素和相关的 **LogoutRequestType** 复杂类型：

```
<element name="LogoutRequest" type="samlp:LogoutRequestType"/>
    <complexType name="LogoutRequestType">
        <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <choice>
                        <element ref="saml:BaseID"/>
                        <element ref="saml:NameID"/>
                        <element ref="saml:EncryptedID"/>
                    </choice>
                    <element ref="samlp:SessionIndex" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
                <attribute name="Reason" type="string" use="optional"/>
                <attribute name="NotOnOrAfter" type="dateTime"
use="optional"/>
            </extension>
        </complexContent>
    </complexType>
<element name="SessionIndex" type="string"/>
```

8.2.7.2 <LogoutResponse>元素

一个<LogoutRequest>消息的接收端必须应答一个<LogoutResponse>消息，类型为 **StatusResponseType**，没有定义附加的内容。

应该通过用来传递消息的协议绑定来对<LogoutResponse>消息进行签字或者进行认证并且提供完整性保护。

接下来的 Schema 段定义了<LogoutResponse>元素：

```
<element name="LogoutResponse" type="samlp:StatusResponseType"/>
```

8.2.7.3 处理规则

消息发送端可以采用 Reason 属性来指示发送<LogoutRequest>的原因。本建议书定义了下面的值，所有的发送端都可以采用；参与者需要针对其他的值达成一致意见：

urn:oasis:names:tc:SAML:2.0:logout:user

因为责任人希望终止指出的会话，这个值定义了正在发送的消息。

urn:oasis:names:tc:SAML:2.0:logout:admin

因为一个管理者希望为责任人终止指出的会话，这个值定义了正在发送的消息。

必须遵守和底层请求和应答消息有关的所有其他处理规则。

在接下来的各节里面提供了额外的处理规则。

1） 会话参与者规则

如果一个会话参与者收到了一个<LogoutRequest>消息，会话参与者必须认证这个消息。如果发送端是授权机构，它提供了一个包含一个链接到责任人当前会话的认证断言的断言，会话参与者必须使通过<saml:BaseID>、<saml:NameID>或者<saml:EncryptedID>元素和这个消息中提供的任何一个<SessionIndex>元素引用的责任人会话无效。如果没有提供<SessionIndex>元素，那么必须使责任人相关的所有会话无效。

会话参与者必须将注销请求消息应用到任何一个满足下述条件的断言，即使断言在注销请求之后到达：

- 断言的主体与<LogoutRequest>中的<saml:BaseID>、<saml:NameID>或者<saml:EncryptedID>元素高度匹配，参见第8.2.3.4节中的定义。

- 一个断言认证断言的SessionIndex属性和注销请求中定义的一个<SessionIndex>元素之一，或者和不包含<SessionIndex>元素的注销请求相匹配。
- 否则，根据断言本身规定的时间条件（任何规定的条件中的NotOnOrAfter属性的值或者主体确认数据），断言有效。

注销请求还没有过期（通过检查消息中的 NotOnOrAfter 属性来决定）。

注 — 这个规则的目的是防止出现一种情况，在这种情况下，一个会话参与者在接收到实际的而且可能仍然有效的断言之前接收到了一个注销请求，该请求的目标是一个单一的或者多个断言（由<SessionIndex>元素标识）。它应该承认这个注销请求直到注销请求被丢弃（请求中的NotOnOrAfter值已经超出）或者已经接收到注销请求的目标断言而且该断言已经得到适当的处理。

2） 会话授权规则

当一个会话授权机构收到了一个<LogoutRequest>消息的时候，会话授权机构必须对发送端进行验证。如果发送端是个会话参与者，会话授权机构给该参与者提供了一个包含一个用于当前会话的认证断言的断言，那么会话授权机构应该按照规定的顺序执行下面的内容：

— 代替会话权威机构所代理的责任人的认证给任何一个会话授权机构发送一个<LogoutRequest>消息，除非有第二个权威机构是<LogoutRequest>的发起者。

— 给每一个会话参与者发送一个<LogoutRequest>消息，会话授权机构在当前的会话中为会话参与者提供断言，而不是由当前<LogoutRequest>的发起者提供断言。

— 通过出现在注销请求消息中的<saml:BaseID>、<saml:NameID>或者<saml:EncryptedID>元素和任何的<SessionIndex>元素来终止责任人的当前会话。

如果会话授权机构成功地终止了责任人的会话，那么它必须向最初的请求端应答一个包含了一个最高级的状态编码为 urn:oasis:names:tc:SAML:2.0:status:Success 的<LogoutResponse>消息。如果它不能这么做，它必须应答一个包含了一个指示差错的最高级状态编码的<LogoutResponse>消息。因此，指示注销操作断言的最高级状态只和会话授权机构本身有关。

会话授权机构应该尝试通过可用的协议绑定去联系每个会话参与者，即使有一次或者多次失败，或者根本无法尝试（例如，因为最初的请求是通过一个协议绑定发生的，它无法将注销信息发送给所有的参与者）。

如果不是所有的会话参与者都成功应答了这些<LogoutRequest>消息（或者无法联系到所有的参与者），那么会话授权机构应该在它的 <LogoutResponse> 消息中包括一个二级状态编码 urn:oasis:names:tc:SAML: 2.0:status:PartialLogout 来指示不是所有的其他会话参与者都通过确认注销而成功地进行了应答。

一个会话授权机构可能由于某些原因而不是因为从一个会话参与者那收到了一个<LogoutRequest>而发起一个注销消息，这些原因如下所示（但是不只限于这些）：

- 如果通过带外方式和一个单独的会话参与者就超时达成了一致，会话授权机构可能会只给这个单独的参与者发送一个<LogoutRequest>。
- 一个达成协议的全局定时周期已经超时。
- 责任人或者某些可信任实体可以直接在会话授权机构处请求责任人的注销。
- 会话授权机构已经决定就责任人的信任状情况达成了一致。

在构建一个注销请求消息的时候，会话授权机构必须将消息的 NotOnOrAfter 属性的值设置为一个时间值，来给消息指定一个过期的时间，在这个时间以后，接收端可能会丢弃注销的请求。这个值应该设为等于或者大于最近作为目标会话一部分发布的（注销请求中，由 SessionIndex 属性来指示）断言中定义的任何一个 NotOnOrAfter 属性值的时间值。

除了第 8.2.6.3 节中为 Reason 属性定义的值，会话授权机构也可以仅采用下面的值：

urn:oasis:names:tc:SAML:2.0:logout:global-timeout

定义了因为已经超出了全局会话超时间隔周期而发送的消息。

urn:oasis:names:tc:SAML:2.0:logout:sp-timeout

定义了因为已经超过了在一个参与者与会话授权机构之间达成一致的超时间隔周期而发送的消息。

8.2.8 名字标识符映射协议

当和一个身份提供者共享一个责任人的标识符的实体希望为同样的责任人获得一个具有特定格式或者联合命名空间的名字标识符，它可以通过此协议向身份提供者发送一个请求。

例如，希望和另外一个它们之间没有共享用于责任人的标识符，服务提供商进行通信的服务提供商， 可以利用和两个服务提供商都共享标识符的身份提供者把它自己的标识符映射到一个新的标识符，通常需要加密的标识符，通过该标识符，它可以和第二个服务提供商进行通信。

不考虑包含到的标识符的类型，应该把映射后的标识符加密到<saml:EncryptedID>元素中，但是如果一个特定的部署方案说明没有必要采用这样的保护方式。

8.2.8.1 <NameIDMappingRequest>元素

为了从身份提供者处为一个责任人请求一个可替换的名字标识符，一个请求端需要发送一个<NameIDMappingRequest>消息。这个消息具有复杂类型 **NameIDMappingRequestType**，它扩展了 **RequestAbstractType** 并且增加了下面的元素：

— <saml:BaseID>或者<saml:NameID>或者<saml:EncryptedID> [必需]

请求端和响应端当前可以识别定义了责任人的标识符和相关的描述性数据（需要关于这个元素的更多信息，参见第 8.1.2 节）。

— <NameIDPolicy> [必需的]

返回的关于标识符的格式和可选的名字限定词的需求。

应该通过用来传递消息的协议绑定来对消息进行签字或者进行认证并且提供完整性保护。

接下来的 Schema 段定义了<NameIDMappingRequest>元素及其 **NameIDMappingRequestType** 复杂类型：

```
<element name="NameIDMappingRequest"
type="samlp:NameIDMappingRequestType"/>
<complexType name="NameIDMappingRequestType">
       <complexContent>
              <extension base="samlp:RequestAbstractType">
                     <sequence>
                            <choice>
                                   <element ref="saml:BaseID"/>
                                   <element ref="saml:NameID"/>
                                   <element ref="saml:EncryptedID"/>
                            </choice>
                            <element ref="samlp:NameIDPolicy"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
```

8.2.8.2 <NameIDMappingResponse>元素

<NameIDMappingRequest>消息的接收端必须应答一个<NameIDMappingResponse>消息。这个消息具有复杂类型 **NameIDMappingResponseType**，它对 **StatusResponseType** 进行了扩展，并且增加了下面的元素：

— <saml:NameID> 或者<saml:EncryptedID> [必需的]

在请求的方式中，标识符和相关属性所定义的责任人通常采用加密的格式。（需要关于此元素的更多的信息，参见第 8.1.2 节）。

用来传输这个消息协议绑定应该对该消息进行签署或者进行认证并且进行完整性保护。

下面的 Schema 段定义了<NameIDMappingResponse>元素及其 **NameIDMappingResponseType** 复杂类型：

```
<element name="NameIDMappingResponse"
type="samlp:NameIDMappingResponseType"/>
<complexType name="NameIDMappingResponseType">
       <complexContent>
              <extension base="samlp:StatusResponseType">
                     <choice>
                            <element ref="saml:NameID"/>
                            <element ref="saml:EncryptedID"/>
                     </choice>
              </extension>
       </complexContent>
</complexType>
```

8.2.8.3 处理规则

如果响应端无法识别请求中标识的责任人，它可以应答一个错误<Status>消息，该消息包含值为 urn:oasis:names:tc:SAML:2.0:status:UnknownPrincipa 的二级<StatusCode>。

在响应端看来，可能通过返回 urn:oasis:names:tc:SAML:2.0:status:InvalidNameIDPolicy 状态码来指示无法或者不愿意通过请求的格式或者命名空间来提供标识符。

必须遵守所有其他的和下面的请求和应答消息有关的处理程序。

8.3 SAML版本

SAML 系列建议书有两个独立的版本。下面将讨论这两个版本，讨论对版本不同进行探测和处理的规则，并给出指导方针以指导 SAML 特定版本信息何时和怎样进行修订。

当版本信息表示为主要和次要版本时，用 *Major.Minor* 表示，当且仅当($Major_B$ > $Major_A$) OR (($Major_B$ = $Major_A$) AND ($Minor_B$ > $Minor_A$))时，版本 $Major_B.Minor_B$ 比版本 $Major_A.Minor_A$ 高。

8.3.1 SAML规范系列版本

SAML 建议书的每次发布都会包含一个主要和次要版本，用以描述建议书的早期版本和后期版本之间的关系。建议书的版本信息会在建议中标明。对建议书的大小和范围的任何改变都需要非正式地指明这些变化是否构成了一个主要或者次要版本。一般来说，如果累积的变化后向兼容早期版本，那么新版本将是一个次要版本。否则，这些变化就构成一个主要版本。

本建议版本为 V2.0。

8.3.1.1 Schema版本

作为一个非标准化文件机制，所有作为规范系列的一部分出版的 XML Schema 文档都会在<xs:schema>元素中包含版本属性，属性值的形式为 *Major.Minor*。版本属性反映出版的规范系列的版本。Schema 版本的属性可以用来区分 Schema 的哪个版本是有效的，或者是否支持同一个逻辑 Schema 的多个版本。

8.3.1.2 SAML断言版本

SAML <Assertion>元素包含采用 *Major.Minor* 形式的字符串表示断言主要和次要版本的属性。SAML 规范系列的每个版本都会解释、说明相同版本断言的句法、语意和处理规则。也就是，规范系列版本 1.0 描述断言版本 1.0，依此类推。

断言版本和为该断言版本指定的 Schema 定义的目标 XML 命名空间之间明显没有关系。

以下是处理规则：

- SAML断言方一定不能发布*Major.Minor*断言版本号为SAML权威所不支持的断言。
- SAML信任方一定不处理主要断言版本号为信任方所不支持的断言。
- 对于次要断言版本号高于信任方支持的次要断言版本号的断言，SAML信任方可能处理也可能拒绝这个断言。但是，共享主要断言版本号的所有断言必须共享相同的一般处理规则和语义，并且可能在实现中采用统一的方法进行处理。例如，如果版本为V1.1的断言使用版本为V1.0的断言的句法，那么这个断言将会被当做V1.0的断言来处理，这样不会出现什么问题。

8.3.1.3 SAML协议版本

SAML 协议的各种请求和响应元素都包含一个表示请求和响应消息的主要和次要版本的属性，该属性采用 *Major.Minor* 形式的字符串表示。SAML 规范系列的每个版本都会解释、说明相同版本协议消息的句法、语义和处理规则。也就是说，规范系列版本 V 1.0 描述请求和响应版本 V 1.0，依此类推。

协议版本和为该协议版本指定的 Schema 定义的目标 XML 命名空间明显没有关系。

SAML 协议的请求和响应元素中所使用的版本号与 SAML 规范系列的特定修订版相对应。

1） 请求版本

以下是请求的处理规则：

- SAML请求端发布的请求版本应是SAML请求端和响应端都支持的最高请求版本。
- 如果SAML请求端不知道SAML响应端的能力，那么它应假定响应端支持请求端所支持的最高请求版本。
- SAML请求端一定不能发布*Major.Minor*请求消息版本对应的响应版本号为请求端不支持的请求消息。
- SAML响应端必须拒绝主要请求版本号为响应端所不支持的任何请求。

对于次要请求版本号高于响应端所支持的最高请求版本号的请求，SAML 响应端可能处理也可能拒绝这个请求。但是，所有共享同一主要版本号的请求必须共享相同的一般处理规则和语义，并且可能在执行中采用统一的方法进行处理。也就是说，如果版本为 V1.1 的请求使用版本为 V1.0 的请求的句法，那么这个请求将会被当做 V1.0 的请求来处理，不会出现什么问题。

2） 响应版本

以下是响应的处理规则：

- SAML响应端一定不能发布响应版本号高于相应请求消息的请求版本号的响应消息。
- 除非报告错误`urn:oasis:names:tc:SAML:2.0:status:RequestVersionTooHigh`，否则SAML响应端一定不能发布主要响应版本号比对应请求消息的主要请求版本号低的响应消息。
- 响应SAML协议版本不相容的错误必须生成顶级值`urn:oasis:names:tc:SAML:2.0:status: VersionMismatch<StatusCode>`，也可以生成报告以下的二级值之一：
- `urn:oasis:names:tc:SAML:2.0:status:RequestVersionTooHigh`；
- `urn:oasis:names:tc:SAML:2.0:status:RequestVersionTooLow`, 或
- `urn:oasis:names:tc:SAML:2.0:status:RequestVersionDeprecated`。

3） 允许的版本组合

当 SAML 断言命名空间的输入允许 SAML 协议 Schema 时，特定的主要版本断言只出现在相同主要版本的响应消息中。例如，如果相应的断言 Schema 在命名空间输入中被引用，那么版本为 V1.1 的断言可以出现在版本 V1.0 的响应消息中，版本为 V1.0 的断言可以出现在版本 V1.1 的响应消息中。但是，版本 V1.0 的断言一定不能出现在版本为 V2.0 的响应消息中，因为它们分属于不同的主要版本。

8.3.2 SAML命名空间版本

XML Schema 文档作为规范系列的一部分出版，它包含一个或多个目标命名空间，命名空间中放置了类型、元素和属性定义。每个命名空间都各不相同，它们以简略的方式表示它们所组成的规范部分的结构和句法定义。

规范系列定义的命名空间 URI 参考通常在 URI 中包含了 *Major.Minor* 形式的版本信息。URI 的主次版本必须与最先引入命名空间和定义命名空间的规范的主次版本相符合。版本信息不仅提供给那些对命名空间进行不透明处理的 XML 处理器，而且还传达规范和规范定义的命名空间之间的关系。第 8.7 节中列出基于 URI 的 SAML 定义的标识符也会讨论这部分内容。

一般来说，由规范的主修订版本定义的命名空间和相关的 Schema 在规范的次修订版本中可以保持有效和稳定。必要时，也可以引入新的命名空间替代旧命名空间，但这种情况最好不要出现。在这种情况下,当引入新命名空间替代旧命名空间时，旧命名空间和它们的相关定义一直到主版本规范修订完成才会失效。

一般情况下，增加或者改变 Schema 内容时，保持命名空间的稳定性是一个重要目标。当某种设计策略致使 Schema 内容容易发生改变时，很难预知这些改变会给旧的实现带来什么样的影响，前向兼容性很难达到。为了命名空间的稳定性，对次修订版本进行这些改变的权力是保留的。特殊情况下例外（例如，为了改正主版本的不足或者修正错误），可以在次修订版本中发生前向兼容的 Schema 改变，使与旧 Schema 不符的新消息生效。

新的扩展和新的消息类型应该按照它们相应的处理规则来进行处理。次修订版本可以引入本建议书描述的会影响到可扩展性的新类型。当关联中指明为强制语义时，旧的实现应该拒绝这样的扩展。例子包括新的询问、断言和条件类型。

8.4 SAML和XML签字句法和处理

SAML 断言和 SAML 协议请求响应消息可能会被签字。签字具有以下的好处：由断言方签字的断言支持断言的完整性、支持 SAML 信任方对断言方的验证。如果签字基于 SAML 权威的公私密钥对的话，还支持来源的不可否认性。

SAML 中不总是需要数字签字。举个例子，在某些情况下签字可以“继承”，比如一个未签字的断言得到其包含的协议响应消息的签字的保护。当被包含的对象（例如断言）有长生命周期时，“继承”签字应当小心使用，原因是为了通过确认，必须保留完整的关联，这样既暴露了 XML 的内容，又增加了潜在的不必要的开销。再举个例子，SAML 信任方或 SAML 请求端可能已直接（无中间物）通过安全信道获得了一个来自 SAML 断言方或者 SAML 响应端的断言或协议消息,断言方或 SAML 响应端就使用数字签字外的其他方法对信任方或 SAML 响应端进行验证。

“直接”认证和两个团体间安全信道的建立有多种不同的技术。包括 TLS、HMAC、基于密码的机制等等。此外，可应用的安全需求还取决于通信应用和断言或传输消息的种类。建议在所有其他的关联中，断言、请求和响应消息都使用数字签字：

— 一个来自非SAML断言方，被SAML信任方所包含的SAML断言应该由SAML断言方签字。

— 一个来自非源发送端，到达目的地的SAML协议消息应该由发送端签字。

— 协议子集可能会规定可选的签字机制，诸如：S/MIME或包含SAML文档的已签字Java对象。要注意保留关联和互动性的应用。当XML签字被定为SAML的主要签字机制时，本建议书会尽量保证它同需要其他签字机制的协议子集的兼容性。

— 除非协议子集规定了可选的签字机制，否则任何XML数字签字都必须被封装。

8.4.1 签字断言

所有的 SAML 断言都可能使用 XML 签字来签字。第 8 节描述的断言 Schema 中会有所涉及。

8.4.2 请求/响应签字

所有的 SAML 协议请求和响应消息都可能使用 XML 签字来签字。附件 A 中的 Schema 会有所涉及。

8.4.3 签字继承

SAML 断言可能会被嵌在另一个 SAML 元素中，例如一个可能被签字的封装<Assertion>、请求或响应。当 SAML 断言不包含<ds:Signature>元素，但却被包含在一个包含<ds:Signature>元素的封装 SAML 元素中，并且签字应用于<Assertion>元素及其所有子元素时，断言可以被认为继承来自封装元素的签字。这样继承签字的结果等效于断言本身被相同的密钥和签字可选项所签字。

许多 SAML 使用 SAML XML 数据，这些数据封装在其他受保护的数据结构中，如签字的 SOAP 消息，S/MIME 包和认证过的 TLS 连接。SAML 协议子集可以定义附加规则使 SAML 元素使用继承签字或者来自关联的其他验证信息，但是除非协议子集特别标明，否则不能推断继承性。

8.4.4 XML签字协议子集

W3X XML Signature:2002 为签字数据的通用 XML 句法提供了灵活性和多种选择。本节详述对这些便利性所作的约束，在这些约束下 SAML 处理器不必处理 XML 签字处理的全部责任人。这样使得 **xs:ID**-typed 属性的特定用法出现在使用签字的根元素上，尤其是<Assertion>和各种请求、响应一般性的 ID 属性。这些属性在本节中全部称为标识符属性。

本协议子集仅应用于在 SAML 断言、请求和响应中直接发现的<ds:Signature>元素的使用。在其中的其他协议子集中签署出现在其他位置但应用于 SAML 内容自用规定其他方法。

8.4.4.1 签字格式和运算法则

在 XML 签字中，签字和文档的关系有三种：封装、已封装和分离。

在签字断言和协议消息时，SAML 断言和协议必须使用已封装的签字。SAML 处理器必须支持 RSA 签字和公钥运算的验证，这种验证同 http://www.w3.org/2000/09/xmldsig# rsa-sha1 中第 6.4 节标明的算法一致。

8.4.4.2 参考

SAML 断言和协议消息必须提供被签字的断言或协议消息中根元素的 ID 属性值。断言或协议消息的根元素可能是也可能不是包含已签字断言或协议消息的实际 XML 文档的根元素（例如，它可能被包含在 SOAP 封装中）。

签字必须包含一个<ds:Reference>元素。对于已签字断言或者协议消息的根元素 ID 属性值来说，<ds:Reference>包含相同文档参考。例如，如果 ID 属性值是“foo”，那么<ds:Reference>元素中的 URI 属性必须是“#foo”。

8.4.4.3 规范方法

SAML 实现应当使用，有或者没有注释的，Exclusive Canonicalization，作为<ds:Transform>算法，它们在<ds:SignedInfo>的<ds:CanonicalizationMethod>元素中。Exclusive Canonicalization 的使用可以保证在 SAML 消息上的签字可以独立于 XML 关联被验证，签字创建在封装于 XML 中的关联。

8.4.4.4 转换

SAML 消息中的签字不必包含转换，除了被封装签字转换（标识符 http://www.w3.org/2000/09/xmldsig#enveloped-signature ）或独立规范转换（标识符 http://www.w3.org/2001/10/xml-exc-c14n# 或 http://www.w3.org/2001/10/xml-exc-c14n#With Comments）。

签字的验证者可能会当做非法来拒绝包含其他转换算法的签字。如果验证者没有拒绝，那么它应该保证 SAML 消息的全部内容都被签字。这种保证可以通过建立哪些转换可接受的带外协议，或者通过手动转换内容并对相同 SAML 消息组成的转换结果进行再次验证来实现。

8.4.4.5 KeyInfo

W3C 签字定义了<ds:KeyInfo>元素的用法。SAML 不需要使用<ds:KeyInfo>，也不用对它的用法强加任何限制。因此，<ds:KeyInfo>可以不出现。

8.4.4.6 例子

下面是包含已签字断言的已签字响应的例子。增加了断行以增强可读性；签字不合法并不能成功通过验证。

```
<Response
  IssueInstant="2003-04-17T00:46:02Z" Version="2.0"
  ID="_c7055387-af61-4fce-8b98-e2927324b306"
  xmlns="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
       <saml:Issuer>https://www.opensaml.org/IDP"</saml:Issuer>
       <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <ds:SignedInfo>
                     <ds:CanonicalizationMethod

       Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                     <ds:SignatureMethod

       Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                     <ds:Reference
URI="#_c7055387-af61-4fce-8b98-e2927324b306">
                            <ds:Transforms>
                                   <ds:Transform

       Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
                                   <ds:Transform

       Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                                          <InclusiveNamespaces
PrefixList="#default saml ds xs xsi"

       xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                                   </ds:Transform>
                            </ds:Transforms>
                            <ds:DigestMethod

       Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>

       <ds:DigestValue>TCDVSuG6grhyHbzhQFWFzGrxIPE=</ds:DigestValue>
                     </ds:Reference>
              </ds:SignedInfo>
              <ds:SignatureValue>

       x/GyPbzmFEe85pGD3c1aXG4Vspb9V9jGCjwcRCKrtwPS6vdVNCcY5rHaFPYWkf+5

       EIYcPzx+pX1h43SmwviCqXRjRtMANWbHLhWAptaK1ywS7gFgsD01qjyen3CP+m3D
                     w6vKhaqledl0BYyrIzb4KkHO4ahNyBVXbJwqv5pUaE4=
              </ds:SignatureValue>
              <ds:KeyInfo>
                     <ds:X509Data>
                            <ds:X509Certificate>
```

```
      MIICyjCCAjOgAwIBAgICAnUwDQYJKoZIhvcNAQEEBQAwgakxCzAJBgNVBAYTAlVT

      MRIwEAYDVQQIEwlXaXNjb25zaW4xEDAOBgNVBAcTB01hZGlzb24xIDAeBgNVBAoT

      F1VuaXZlcnNpdHkgb2YgV2lzY29uc2luMSswKQYDVQQLEyJEaXZpc2lvbiBvZiBJ

      bmZvcm1hdGlvbiBUZWNobm9sb2d5MSUwIwYDVQQDExxIRVBLSSBTZXJ2ZXIgQ0Eg

      LS0gMjAwMjA3MDFBMB4XDTAyMDcyNjA3Mjc1MVoXDTA2MDkwNDA3Mjc1MVowgYsx

      CzAJBgNVBAYTAlVTMREwDwYDVQQIEwhNaWNoaWdhbjESMBAGA1UEBxMJQW5uIEFy

      Ym9yMQ4wDAYDVQQKEwVVQ0FJRDEcMBoGA1UEAxMTc2hpYjEuaW50ZXJuZXQyLmVk

      dTEnMCUGCSqGSIb3DQEJARYYcm9vdEBzaGliMS5pbnRlcm5ldDIuZWR1MIGfMA0G

      CSqGSIb3DQEBAQUAA4GNADCBiQKBgQDZSAb2sxvhAXnXVIVTx8vuRay+x50z7GJj

      IHRYQgIv6IqaGG04eTcyVMhoekE0b45QgvBIaOAPSZBl13R6+KYiE7x4XAWIrCP+

      c2MZVeXeTgV3Yz+USLg2Y1on+Jh4HxwkPFmZBctyXiUr6DxF8rvoP9W7O27rhRjE

      pmqOIfGTWQIDAQABox0wGzAMBgNVHRMBAf8EAjAAMAsGA1UdDwQEAwIFoDANBgkq

      hkiG9w0BAQQFAAOBgQBfDqEW+OI3jqBQHIBzhujN/PizdN7s/z4D5d3pptWDJf2n

      qgi7lFV6MDkhmTvTqBtjmNk3No7v/dnP6Hr7wHxvCCRwubnmIfZ6QZAv2FU78pLX

      8I3bsbmRAUg4UP9hH6ABVq4KQKMknxu1xQxLhpR1ylGPdiowMNTrEG8cCx3w/w==
                        </ds:X509Certificate>
                  </ds:X509Data>
            </ds:KeyInfo>
      </ds:Signature>
      <Status>
            <StatusCode
Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
      </Status>
      <Assertion ID="_a75adf55-01d7-40cc-929f-dbd8372ebdfc"
            IssueInstant="2003-04-17T00:46:02Z" Version="2.0"
            xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
            <Issuer>https://www.opensaml.org/IDP</Issuer>
            <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                  <ds:SignedInfo>
                        <ds:CanonicalizationMethod
                              Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#"/>
                        <ds:SignatureMethod

      Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                        <ds:Reference URI="#_a75adf55-01d7-40cc-929f-
dbd8372ebdfc">
                              <ds:Transforms>
                                    <ds:Transform

      Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
                                    <ds:Transform

      Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                                          <InclusiveNamespaces
                                                PrefixList="#default
saml ds xs xsi"

      xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                                    </ds:Transform>
                              </ds:Transforms>
                              <ds:DigestMethod

      Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>

      <ds:DigestValue>Kclet6XcaOgOWXM4gty6/UNdviI=</ds:DigestValue>
```

```
                        </ds:Reference>
                    </ds:SignedInfo>
                    <ds:SignatureValue>

      hq4zk+ZknjggCQgZm7ea8fI79gJEsRy3E8LHDpYXWQIgZpkJN9CMLG8ENR4Nrw+n

      7iyzixBvKXX8P53BTCT4VghPBWhFYSt9tHWu/AtJfOTh6qaAsNdeCyG86jmtp3TD
                        MwuL/cBUj2OtBZOQMFn7jQ9YB7klIz3RqVL+wNmeWI4=
                    </ds:SignatureValue>
                    <ds:KeyInfo>
                        <ds:X509Data>
                            <ds:X509Certificate>

      MIICyjCCAjOgAwIBAgICAnUwDQYJKoZIhvcNAQEEBQAwgakxCzAJBgNVBAYTAlVT

      MRIwEAYDVQQIEwlXaXNjb25zaW4xEDAOBgNVBAcTB01hZGlzb24xIDAeBgNVBAoT

      F1VuaXZlcnNpdHkgb2YgV2lzY29uc2luMSswKQYDVQQLEyJEaXZpc2lvbiBvZiBJ

      bmZvcm1hdGlvbiBUZWNobm9sb2d5MSUwIwYDVQQDExxIRVBLSSBTZXJ2ZXIgQ0Eg

      LS0gMjAwMjA3MDFBMB4XDTAyMDcyNjA3Mjc1MVoXDTA2MDkwNDA3Mjc1MVowgYsx

      CzAJBgNVBAYTAlVTMREwDwYDVQQIEwhNaWNoaWdhbjESMBAGA1UEBxMJQW5uIEFy

      Ym9yMQ4wDAYDVQQKEwVVQ0FJRDEcMBoGA1UEAxMTc2hpYjEuaW50ZXJuZXQyLmVk

      dTEnMCUGCSqGSIb3DQEJARYYcm9vdEBzaGliMS5pbnRlcm5ldDIuZWR1MIGfMA0G

      CSqGSIb3DQEBAQUAA4GNADCBiQKBgQDZSAb2sxvhAXnXVIVTx8vuRay+x50z7GJj

      IHRYQgIv6IqaGG04eTcyVMhoekE0b45QgvBIaOAPSZBl13R6+KYiE7x4XAWIrCP+

      c2MZVeXeTgV3Yz+USLg2Y1on+Jh4HxwkPFmZBctyXiUr6DxF8rvoP9W7O27rhRjE

      pmqOIfGTWQIDAQABox0wGzAMBgNVHRMBAf8EAjAAMAsGA1UdDwQEAwIFoDANBgkq

      hkiG9w0BAQQFAAOBgQBfDqEW+OI3jqBQHIBzhujN/PizdN7s/z4D5d3pptWDJf2n

      qgi7lFV6MDkhmTvTqBtjmNk3No7v/dnP6Hr7wHxvCCRwubnmIfZ6QZAv2FU78pLX

      8I3bsbmRAUg4UP9hH6ABVq4KQKMknxu1xQxLhpR1ylGPdiowMNTrEG8cCx3w/w==
                            </ds:X509Certificate>
                        </ds:X509Data>
                    </ds:KeyInfo>
                </ds:Signature>
                <Subject>
                    <NameID
                        Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:emailAddress">
                        scott@example.org
                    </NameID>
                    <SubjectConfirmation
                        Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"/>
                </Subject>
                <Conditions NotBefore="2003-04-17T00:46:02Z"
                        NotOnOrAfter="2003-04-17T00:51:02Z">
                    <AudienceRestriction>
                        <Audience>http://www.opensaml.org/SP</Audience>
                    </AudienceRestriction>
                </Conditions>
                <AuthnStatement AuthnInstant="2003-04-17T00:46:00Z">
                    <AuthnContext>
                        <AuthnContextClassRef>

      urn:oasis:names:tc:SAML:2.0:ac:classes:Password
                        </AuthnContextClassRef>
                    </AuthnContext>
                </AuthnStatement>
        </Assertion>
</Response>
```

8.5 SAML和XML加密句法和处理

加密是实现机密性的方法。机密性的最普遍的目的是保护个体的私密性，或为了组织的利益和类似原因而保护组织的秘密。保证一些其他安全机制的有效性可能也需要机密性。例如，一个密码或者密钥或许需要被加密。

为了保护全部或部分 SAML 断言的机密性，提供了以下几种方法用于加密：

- 通信的机密性可以通过与一种特别的绑定或者协议子集机制相关联来实现。例如，SOAP绑定支持使用TLS（见IETF RFC 2246）或SOAP消息机密安全机制。
- 使用 <SubjectConfirmationData> 中的 <ds:KeyInfo> 元素可以保护 <SubjectConfirmation>秘密，允许密钥或其他秘密被加密。
- 一个完整的<Assertion> 元素可能会如第8.1.3.4节中描述的那样被加密。
- <BaseID>或<NameID>元素可能会如第8.1.2.4节中描述的那样被加密。
- <Attribute>元素可能会如第8.1.7.3.2节中描述的那样被加密。

8.5.1 一般的考虑

使用 XML 加密可以提供对<Assertion>、<BaseID>、<NameID> 和 <Attribute>元素的加密。被加密的数据和一个或多个被加密的密钥必须取代 XML 实例中相同位置的纯文本信息。如果<EncryptedData>元素出现，那么应当使用它的类型属性，必须有 http://www.w3.org/2001/04/xmlenc#Element 值。

注（资料性的）— PE30（见 OASIS PE:2006）建议用0或更多替代第二行中的1或更多。

任何使用 XML 加密来定义的使用算法都可能会被用于完成加密。定义了 SAML Schema，以便加密数据的包含物生成一个合法的实例。

8.5.2 联合签字和加密

XML 加密和 XML 签字可能会被用于联合。当一个断言需要签字和加密时，应用以下所述规则。信任方必须完成签字的确认和解密，这与签字和加密的目的相反。

- 当一个签字的<Assertion>元素被加密，必须首先计算签字，并且在<Assertion>元素被加密之前放置进去。
- 当<BaseID>、<NameID>或<Attribute> 元素被加密，必须先进行加密，然后再在包含加密元素的断言或消息上计算签字。

8.6 SAML扩展性

SAML 支持多种扩展，包括扩展断言和协议 Schema。如何定义新的协议子集，结合扩展以促进 SAML 框架的新用途，见本建议书的 SAML 协议子集条款。

8.6.1 Schema扩展

SAML Schema 中的元素不能置换，这意味着没有 SAML 元素能像置换组的头元素那样服务。但是，SAML 类型并不是不能改变，所有的 SAML 类型都可以扩展和限制。实际上，这意味着扩展只是特指类型而不是元素，下面条款仅讨论为支持扩展已经特别设置的元素和类型。通过 xsi:type 属性包含在 SAML 实例中。

8.6.1.1 断言Schema扩展

如果断言包或断言包所包含状态的任何一个部分使用了扩展机制，SAML 断言 Schema 允许对它们进行分开处理。

扩展 Schema 中的扩展点尤其注重以下元素的使用；它们的类型设置为抽象类型，这样元素类型就只作为派生类型的基类来用。

- <BaseID>和 **BaseIDAbstractType**
- <Condition>和 **ConditionAbstractType**

- <Statement>和**StatementAbstractType**

以下结构可直接用做 SAML 的一部分，是扩展尤其注重的目标。

- <AuthnStatement>和**AuthnStatementType**
- <AttributeStatement>和**AttributeStatementType**
- <AuthzDecisionStatement>和**AuthzDecisionStatementType**
- <AudienceRestriction>和**AudienceRestrictionType**
- <ProxyRestriction>和**ProxyRestrictionType**
- <OneTimeUse>和**OneTimeUseType**

8.6.1.2 协议Schema扩展

扩展 Schema 中的扩展点尤其注重以下 SAML 协议元素的使用；协议元素设置为抽象类型，这样元素类型就仅作为派生类型的基类来用。

- <Request>和**RequestAbstractType**
- <SubjectQuery>和**SubjectQueryAbstractType**

以下结构可直接用做 SAML 的一部分，是扩展尤其注重的目标

- <AuthnQuery>和**AuthnQueryType**
- <AuthzDecisionQuery>和**AuthzDecisionQueryType**
- <AttributeQuery>和**AttributeQueryType**
- **StatusResponseType**

8.6.2 Schema通配符扩展点

SAML Schema 在某些位置使用通配符结构，以允许来自任意命名空间的元素和属性的使用，如同一个不需要扩展 Schema 的内置扩展点。

8.6.2.1 断言扩展点

断言 Schema 中的下列结构允许从结构中的任意命名空间进行构造：

- <SubjectConfirmationData>：使用**xs:anyType**，允许任何子元素和属性。
- <AuthnContextDecl>：使用**xs:anyType**，允许任何子元素和属性。
- <AttributeValue>：使用**xs:anyType**，允许任何子元素和属性。
- <Advice>和**AdviceType**：除了本地 SAML元素，允许来自其他不严格Schema确认过程的命名空间的元素。

断言 Schema 中的以下结构允许任意的全局属性：

- <Attribute>和**AttributeType**

8.6.2.2 协议扩展点

协议 Schema 中的以下结构允许从结构中的任意命名空间进行构造：

- <Extensions>和**ExtensionsType**：允许来自其他不严格Schema确认过程的命名空间的元素。
- <StatusDetail>和**StatusDetailType**：允许来自其他不严格Schema确认过程的命名空间的元素。
- <ArtifactResponse>和**ArtifactResponseType**：允许来自其他不严格Schema确认过程的命名空间的元素。（然而,尤其是传送一个SAML请求或者响应消息）

8.6.3 标识符扩展

SAML 使用基于 URI 的标识符有许多目的，比如状态码和名字标识符格式。并且定义了一些可能用于这些目的的标识符，它们大多在第 8.7 节中列出。但是，SAML 还可能定义其他的出于这些目的的基于 URI 标识符。建议这些其他的标识符在正式的使用协议子集中定义。在任何情况下一个特定的当做标识符使用的 URI 的意义不能发生改变，也不能被用于表示两个不同的事情。

8.7 已定义的SAML标识符

以下子条款为公共资源接入活动定义了基于 URI 的标识符，主体名字标识符格式和属性名格式。

一个现有的 URN 可能会被用于规定一个协议。在 IETF 协议中，使用协议的最新 RFC 的 URN。根据首次被引用的规范版本，专为 SAML 创建的 URI 引用有些以下的 stem：

```
urn:oasis:names:tc:SAML:1.0:
urn:oasis:names:tc:SAML:1.1:
urn:oasis:names:tc:SAML:2.0:
```

本建议书引用最新的 stem。

8.7.1 动作命名空间标识符

下列标识符可能会用在<Action>属性的 Namespace 属性中，用以标识在资源上完成的一公共系列动作。

8.7.1.1 Read/Write/Execute/Delete/Control

URI: urn:oasis:names:tc:SAML:1.0:action:rwedc

定义的动作：Read Write Execute Delete Control

这些动作的说明如下：

Read：主体可以读资源。

Write：主体可以修改资源。

Execute：主体可以执行资源。

Delete：主体可以删除资源。

Control：主体可以规定资源的接入控制策略。

8.7.1.2 否定的Read/Write/Execute/Delete/Control

URI: urn:oasis:names:tc:SAML:1.0:action:rwedc-negation

定义的动作：Read Write Execute Delete Control ~Read ~Write ~Execute ~Delete ~Control

第 8.7.1.1 节规定的动作用那里所描述的方式说明。动作前加前缀（～）表示否定允许，用于确指规定的允许被拒绝。这样，一个描述为被认证的主体执行动作~Read 即肯定地表示它被拒绝读许可。

SAML 权威一定不能授权通过动作和它的否定格式。

8.7.1.3 Get/Head/Put/Post

URI: urn:oasis:names:tc:SAML:1.0:action:ghpp

定义的动作：GET HEAD PUT POST

这些动作与相应的 HTTP 操作绑定。例如，一个被授权的主体在资源上执行 GET 动作，它被授权检索这个资源。

GET 和 HEAD 动作松散对应约定的读允许，PUT 和 POST 动作对应写允许。然而，这种对应并不是严格的，因为 HTTP GET 操作可能会导致数据被修改，而 POST 操作可能会导致请求中指定之外的资源被修改。由于这个原因，给出了一个独立的动作 URI 引用。

8.1.7.4 UNIX文件允许

URI: urn:oasis:names:tc:SAML:1.0:action:unix

定义的动作是用数字（八进制）符号表示的一系列 UNIX 文件接入允许。

动作字符串是一个四位数的数字代码：

extended user group world

这里 *extended* 接入允许具有值：

+2，如果 sgid 置位

+4，如果 suid 置位

user group 和 *world* 接入允许具有值：

+1，如果 execute 允许被获准

+2，如果 write 允许被获准

+4，如果 read 允许被获准

例如，0754 表示 UNIX 文件接入允许：用户可读、写、执行；组可读、执行；公众可读。

8.7.2 属性名字格式标识符

以下标识符可能用于定义在 **AttributeType** 复杂类型上规定的 `NameFormat` 属性中，为解释命名而指明属性名的分类。

8.7.2.1 未明确的

URI: `urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified`

这个属性名的解释在单独实现中解释。

8.7.2.2 URI引用

URI: `urn:oasis:names:tc:SAML:2.0:attrname-format:uri`

属性名遵循 URI 引用惯例，例如当属性名用于 XACML 属性标识符。URI 内容或者命名 Schema 的解释是应用特定的。见第 11 节使用这个标识符的属性协议子集。

8.7.2.3 基础

URI: `urn:oasis:names:tc:SAML:2.0:attrname-format:basic`

可作为属性名的字符串类必须从属于原语类型 **xs:Name** 的值集合中选取，原语类型在 W3C XML Datatypes，第 3.3.6 节中定义。见 13 节使用这个标识符的属性协议子集。

8.7.3 命名标识符格式标识符

以下标识符可能用于`<NameID>`、`<NameIDPolicy>`或`<Issuer>`元素（见第 8.1.2 节）的 `Format` 属性中，指明元素内容和相关处理规则的一般格式，如果有的话。

注 — SAML V1.1中反对的几个标识符已经在SAML V2.0中被移除。

8.7.3.1 未明确的

URI: `urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified`

元素内容在单独实现中进行解释。

8.7.3.2 email地址

URI: `urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress`

指明元素的内容是以 email 地址为形式的，正如 IETF RFC 2822，第 3.4.1 节中定义的"addr-spec"。addr-spec 具有 local-part@domain 的形式。注意，addr-spec 之前没有短语（诸如一个通用的名字），之后没有注释（圆括号内的文本），也没有被括在"<"和">"内。

8.7.3.3 X.509主体名字

URI: urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName

指明元素内容的形式是 W3C 签字中指定的<ds:X509SubjectName>元素的内容。应该注意的是 W3C XML Signature 为 X.509 主体名字指定了不同于 IETF RFC 2253 中不同的编码规则。

8.7.3.4 Windows域名限定名

URI: urn:oasis:names:tc:SAML:1.1:nameid-format:WindowsDomainQualifiedName

指明元素内容是一个 Windows 域名限定名。Windows 域名限定名用户名称是一个"DomainName\UserName "形式的字符串。域名和"\"分隔符可以被省略。

8.7.3.5 Kerberos责任人名称

URI: urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos

指明元素内容的形式是使用格式 name[/instance]@REALM 的 Kerberos 主体名称。IETF RFC 1510 中描述了允许用于名称、实例、域的句法、格式和字符。

8.7.3.6 实体标识符

URI: urn:oasis:names:tc:SAML:2.0:nameid-format:entity

指明元素的内容是一个实体标识符，提供基于 SAML 的业务（如 SAML 权威、请求端或响应端）或者是一个 SAML 协议子集的参与者（如支持浏览器 SSO 协议子集的服务提供商）。标识符可以在<Issuer>元素来识别 SAML 请求、响应或者断言的发行者，或者可以用在<NameID>中为发行 SAML 请求、响应和断言的系统实体做断言。它还可以用于其他元素和属性，其目的是识别不同协议交换中的系统实体。

实体标识符的句法是一个长度不多于 1024 字符的 URI。建议系统实体使用包含它自己域名的 URL，以识别它自己。

NameQualifier、SPNameQualifier 和 SPProvidedID 属性必须被省略。

8.7.3.7 持续标识符

URI: urn:oasis:names:tc:SAML:2.0:nameid-format:persistent

指明元素的内容对责任人是一个持续不透明的标识符，说明主体是身份提供者、服务提供商或是服务提供商的联盟。由身份提供者生成的持续名称标识符必须通过使用伪随机值来重构，伪随机值与主体的实际标识符（比如用户名）并无直接的相符关系。这样做的目的是为了创建一个非公有的相互匿名机制来保护主体实体或活动的发现。持续命名标识符值一定不能超过 256 字符长度。

如果元素的 NameQualifier 属性出现，那么它必须包含产生标识符的身份提供者的唯一标识（见第 8.7.3.6 节）。如果这个属性值可以从包含元素的消息内容中得到，例如协议消息的发布者或者主体中包含标识符的断言，那么它可以被省略。不同的系统实体可能会滞后发布它自己的协议消息或包含标识符的断言；在这种情况下 NameQualifier 标识符不会改变，但是必须继续识别最初创建标识符的实体（在这种情况下一定不能被省略）。

如果元素的 SPNameQualifier 元素属性出现，那么它必须包含服务提供商或者与身份提供者联盟的唯一标识符（见第 8.7.3.6 节）。如果服务提供商在一个仅用于消费的消息中直接包含了元素，那么元素可以被省略，元素值将作为服务提供商的唯一标识符。

元素的 SPProvidedID 属性必须包含主要部分的可选标识符，这个标识符由服务提供商或者联盟最新设置（见第 8.2.6 节）。如果没有建立这样的标识符，那么这个属性必须省略。

持续标识符用于隐私保护机制，它不能与那些还没有建立共享标识符的提供商在明文中实现共享。此外，它也不能在日志文件或没有相应控制和保护的相似位置中出现。没有隐私保护需求的实现可以任意使用 SAML 交换中的其他标识符，但是一定不能使用持续但透明的值过载这种格式。

当持续标识符典型地用于反映两个提供者之间的账户链接关系时，服务提供商不必识别或者使用持续标识符的长期本性（long term nature），或者建立这样的链接。这种“单边”关系没有明显的不同，不会影响到身份提供者或本建议书定义的协议中的持续标识符的任何处理规则。

NameQualifier 和 SPNameQualifier 属性指明了标识符的生成方法，但没有说明标识符的使用方法。如果持续标识符由一个特定的身份提供者生成，那么 NameQualifier 属性值就会在此时永久确定。如果接收这个标识符的服务提供商具有身份提供者的角色，并发布它自己的包含这个标识符的断言，那么 NameQualifier 属性值不能改变（当然更不能被省略）。它可以选择生成它自己的持续标识符，以表征两个属性值中的重要方和链接关系。这是开发决定。

8.7.3.8 短暂标识符

URI: urn:oasis:names:tc:SAML:2.0:nameid-format:transient

指明元素的内容是一个短暂语义的标识符，信任方应该把它当做一个不透明的、临时的值对待。短暂标识符值必须按照 SAML 标识符（见第 7.4 节）规则来产生，不能超过 256 字符的长度。

NameQualifier 和 SPNameQualifier 属性可能会用于表明标识符是一个短暂的临时的相互匿名标识符。在这种情况下，按照第 8.7.3.7 节的规则，它们可以被省略。

8.7.4 应答标识符

以下标识符可能用于针对 **RequestAbstractType** 和 **StatusResponseType** 复杂类型规定的 Consent 属性，以传达在什么情况下责任人是否为消息给出同意。

8.7.4.1 未明确的

URI: urn:oasis:names:tc:SAML:2.0:consent:unspecified

对责任人 同意没有要求。

8.7.4.2 被获得的

URI: urn:oasis:names:tc:SAML:2.0:consent:obtained

指明责任人的应答已被消息的发布者所获得。

8.7.4.3 优先的

URI: urn:oasis:names:tc:SAML:2.0:consent:prior

指明责任人的同意已被优先于发起消息的消息发布者所获得。

8.7.4.4 隐式的

URI: urn:oasis:names:tc:SAML:2.0:consent:current-implicit

指明责任人的同意在发起消息的时候已经被消息发布者隐式获得，作为同意主要表示的一部分。隐式的同意相比优先应答，比如活动会话部分，更接近及时动作和表达。

8.7.4.5 显式的

URI: urn:oasis:names:tc:SAML:2.0:consent:current-explicit

指明责任人的同意在发起消息的时候已被消息的发布者显式获得。

8.7.4.6 无效的

URI: urn:oasis:names:tc:SAML:2.0:consent:unavailable
指明消息的发布者不获得同意。

8.7.4.7 不适用的

URI: urn:oasis:names:tc:SAML:2.0:consent:inapplicable
指明消息的发布者相信他们不需要获得或者报告同意。

9 SAML元数据

SAML 协议子集要求在涉及标识符的，绑定支持和端点，证书和密钥等系统实体之间达成一致。本节定义了一种针对 SAML 系统实体的可扩展的元数据格式，由反映 SAML 协议子集的角色来组织。这些角色包括 SSO 身份提供者，SSO 服务提供商，联盟，属性授权中心，属性请求端和策略决定点。

9.1 元数据

SAML 元数据围绕着一个可扩展的集合来组织，这个集合的角色代表了系统实体支持的 SAML 协议和协议子集的一般组合。每个角色都通过 RoleDescriptor 的可扩展基类衍生的元素来描述。这些描述符顺序集成为<EntityDescriptor>容器元素，是 SAML 元数据的基本单元。一个实体可以选择性的表示其他实体的联盟，例如服务提供商的联盟。为此目的提供了<AffiliationDescriptor>。

这些描述符可以顺序集成为使用<EntitiesDescriptor>元素的嵌套组。

可以支持多种建立元数据可信度的安全机制，尤其是那些具有对本建议书定义的大部分元素分别签字能力的机制。

当具有父/子关系的元素包含普通属性时，例如缓存或过期信息，父元素优先。

注 — 一般的，SAML元数据不作为对于给定系统实体的容量或选项的权威断言。即，它应该准确，但无需详尽。某个特殊选项的省略不表示该选项是否被支持，而仅仅是没有断言而已。举例来说，一个SAML属性授权中心可以在<AttributeAuthorityDescriptor>中支持任意数量的未命名的属性。省略可能反映了私密的原因或其他考虑因素。相反的，对一个给定的属性表示支持并不说明某个给定的请求端能够或者会接收它。

9.1.1 命名空间

SAML 元数据使用以下的命名空间：

```
urn:oasis:names:tc:SAML:2.0:metadata
```

本建议书使用使用命名空间前缀 md:表示上面的命名空间。

下面的 Schema 段说明了在 SAML 元数据文件中命名空间的用法：

```
<schema
    targetNamespace="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="http://www.w3.org/2000/09/xmldsig#"
        schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-
20020212/xmldsig-core-schema.xsd"/>
    <import namespace="http://www.w3.org/2001/04/xmlenc#"
        schemaLocation="http://www.w3.org/TR/2002/REC-xmlenc-core-
20021210/xenc-schema.xsd"/>
    <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
```

```
        schemaLocation="saml-schema-assertion-2.0.xsd"/>
    <import namespace="http://www.w3.org/XML/1998/namespace"
        schemaLocation="http://www.w3.org/2001/xml.xsd"/>
    <annotation>
        <documentation>
            Document identifier: saml-schema-metadata-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
              V2.0 (March, 2005):
                Schema for SAML metadata, first published in SAML 2.0.
        </documentation>
    </annotation>
…
</schema>
```

9.1.2 通用类型

本节定义几种元数据类型，它们用来定义元素和属性。

9.1.2.1 简单类型entityIDType

简单类型 **entityIDType** 限制 XML 方案数据类型 **anyURI** 的最大长度为 1024 个字符。**entityIDType** 作为 SAML 实体的唯一标识符使用。可参考和 8.7.3.6 节。该类型的标识符在所有实体之间必须是唯一的，这些实体在一个给定的配置中交互作用。使用 URI 并坚持单个 URI 禁止指向不同的实体这一规则满足了这个要求。

下面的 Schema 段说明了 **entityIDType** 简单类型的用法：

```
<simpleType name="entityIDType">
       <restriction base="anyURI">
              <maxLength value="1024"/>
       </restriction>
</simpleType>
```

9.1.2.2 复杂类型EndpointType

复杂类型 **EndpointType** 描述了 SAML 协议绑定端点，在那里 SAML 实体能被发送协议消息。多种协议或特定配置的元数据元素与该类型绑定。它由以下属性组成：

— `Binding` [必需的]

一个必需的属性，指定端点支持的 SAML 绑定。每个映射都分配一个 URI 来标识。

— `Location` [必需的]

一个必需的属性，指定端点的位置。该 URI 的可允许的句法依赖于协议绑定。

— `ResponseLocation` [可选的]

可选的指定一个不同的位置，响应消息作为协议或配置的一部分应该被发送到那里。该 URI 的可允许的句法依赖于协议绑定。

`ResponseLocation` 属性用来指定不同的端点，以接收与协议或配置相关的请求和响应消息，而不是作为负载平衡或冗余的手段（该类型的多种元素是为此目的而包含的）。当一个角色包含一个该类型的元素，而其所属的协议或配置只应用单独的消息类型（请求或响应）时，`ResponseLocation` 属性是不使用的。

注（资料性的）— PE41（参见OASIS PE:2006）给上段文字添加以下语句以作进一步阐明：

如果 `ResponseLocation` 属性被省略，任何与协议或配置相关的响应消息可以假定在位置属性指明的 URI 处被处理。

在多数关联中，本类型的元素在方案中可以无限序列出现。这是为了允许实体在多个端点提供协议或配置，通常还带有不同的协议绑定，使得元数据消费者可以选择它所需要的合适的端点。多个端点也可以提供“客户端”的负载平衡或故障转移，特别是在同步协议绑定的情况下。

该元素也允许使用定义在非 SAML 命名空间中规定的任意元素和属性。任何这样的内容都必须是有效的命名空间。

下面的 Schema 段定义了 **EndpointType** 复杂类型：

```
<complexType name="EndpointType">
      <sequence>
            <any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
      <attribute name="Binding" type="anyURI" use="required"/>
      <attribute name="Location" type="anyURI" use="required"/>
      <attribute name="ResponseLocation" type="anyURI" use="optional"/>
      <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
```

9.1.2.3 复杂类型IndexedEndpointType

复杂类型 **IndexedEndpointType** 用一对属性扩展 **EndpointType**，以允许其他相同的端点建立索引，使得协议消息能够引用它们。它由以下附加属性组成：

— index [必需的]

一个必需的属性，它给端点分配一个唯一的整数型值，使得其能在协议消息中被引用。索引值只需要在类似元素集合中是唯一的，这些类似元素被同一个父元素包含（也就是说，它们不需要在整个实例中唯一）。

— 号 isDefault [可选的]

一个可选的布尔属性，用来指定一组索引集合中的缺省端点。如果省略，其值假定为 false。

在这些基于该类型的类似端点序列中，缺省端点是 isDefault 属性设置为 true 的第一个此类端点。如果没有这样的端点存在，则缺省端点是 isDefault 属性没有被设置为 false 的第一个此类端点。如果这样的端点还不存在，则缺省端点是序列中的第一个元素。

注（资料性的）— PE37（参见OASIS PE:2006）建议如下阐明上段：

在这些共享通用元素名称和命名空间（即一个角色中的所有<md:AssertionConsumerService>实例）的索引端点序列中，缺省端点是 isDefault 属性设置为 true 的第一个此类端点。如果没有这样的端点存在，则缺省端点是 isDefault 属性没有被设置为 false 的第一个此类端点。如果这样的端点还不存在，则缺省端点是序列中的第一个元素。

下面的 Schema 段定义了 **IndexedEndpointType** 复杂类型：

```
<complexType name="IndexedEndpointType">
      <complexContent>
            <extension base="md:EndpointType">
                  <attribute name="index" type="unsignedShort"
use="required"/>
                  <attribute name="isDefault" type="boolean"
use="optional"/>
            </extension>
      </complexContent>
</complexType>
```

9.1.2.4 复杂类型localizedNameType

LocalizedNameType 复杂类型用一个标准的 XML 语言属性扩展取值为字符串的元素。下面的 Schema 段定义了 **localizedNameType** 复杂类型：

```
<complexType name="localizedNameType">
      <simpleContent>
            <extension base="string">
                  <attribute ref="xml:lang" use="required"/>
            </extension>
      </simpleContent>
</complexType>
```

9.1.2.5 复杂类型localizedURIType

LocalizedURIType 复杂类型用一个标准的 XML 语言属性扩展取值为 URI 的元素。

下面的 Schema 段定义了 **localizedURIType** 复杂类型：

```
<complexType name="localizedURIType">
<simpleContent>
              <extension base="anyURI">
                     <attribute ref="xml:lang" use="required"/>
              </extension>
       </simpleContent>
</complexType>
```

9.1.3 根元素

一个 SAML 元数据实例描述单个实体或多个实体。在前一种情况下，根元素必须是<EntityDescriptor>。在后一种情况下，根元素必须是<EntitiesDescriptor>。

9.1.3.1 元素<EntitiesDescriptor>

<EntitiesDescriptor>元素包含一个随意指定的 SAML 实体组的元数据。其 **EntitiesDescriptorType** 复杂类型包含<EntityDescriptor>元素、<EntitiesDescriptor>元素或这二者的序列。

— ID [可选的]

元素档案的唯一性标识符，其典型应用是在签字时作为一个参考点。

— validUntil [可选的]

可选属性，指出包含在元素和任意被包含的元素中的元数据的废止时间。

— cacheDuration [可选的]

可选属性，指出消费方应该缓存包含在元素和任意被包含的元素中的元数据的最大时间长度。

— Name [可选的]

字符串名称，标明某个配置关联中的一组 SAML 实体。

— <ds:Signature> [可选的]

XML 签字，用来鉴别包含的元素及其内容，如第 8 节所述。

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。扩展元素对于一个非 SAML 定义的命名空间来说，必须是有效的命名空间。

— <EntitiesDescriptor>或<EntityDescriptor> [一个或多个]

包含一个或多个 SAML 实体的元数据，或一个额外的元数据的嵌套组。

当用做一个元数据实例的根元素时，该元素必须包含一个 validUntil 或 cacheDuration 属性。建议仅元数据实例的根元素包含其中任一属性。

下面的 Schema 段定义了<EntitiesDescriptor>元素及其 **EntitiesDescriptorType** 复杂类型：

```
<element name="EntitiesDescriptor" type="md:EntitiesDescriptorType"/>
<complexType name="EntitiesDescriptorType">
       <sequence>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="md:Extensions" minOccurs="0"/>
              <choice minOccurs="1" maxOccurs="unbounded">
                     <element ref="md:EntityDescriptor"/>
                     <element ref="md:EntitiesDescriptor"/>
              </choice>
       </sequence>
       <attribute name="validUntil" type="dateTime" use="optional"/>
       <attribute name="cacheDuration" type="duration" use="optional"/>
       <attribute name="ID" type="ID" use="optional"/>
       <attribute name="Name" type="string" use="optional"/>
</complexType>
```

```
<element name="Extensions" type="md:ExtensionsType"/>
<complexType final="#all" name="ExtensionsType">
       <sequence>
              <any namespace="##other" processContents="lax"
maxOccurs="unbounded"/>
       </sequence>
</complexType>
```

9.1.3.2 元素<EntityDescriptor>

<EntityDescriptor>元素规定单个 SAML 实体的元数据。单个实体可以在多种协议子集的支持下担任许多不同的角色。本建议书直接支持以下的具体角色和抽象的<RoleDescriptor>元素作为扩展:

- SSO身份提供者;
- SSO服务提供商;
- 认证授权中心;
- 属性授权中心;
- 策略决定点;
- 联盟。

其 **EntityDescriptorType** 复杂类型由以下元素和属性组成:

— entityID [必需的]

规定 SAML 实体的唯一标识符，实体的元数据由元素的内容进行描述。

— ID [可选的]

元素档案的唯一性标识符，其典型应用是在签字时作为一个参考点。

— validUntil [可选的]

可选属性，指示包含在元素和任意被包含的元素中的元数据的废止时间。

— cacheDuration [可选的]

可选属性，指示消费方应该缓存包含在元素和任意被包含的元素中的元数据的最大时间长度。

— <ds:Signature> [可选的]

XML 签字，用来鉴别包含的元素及其内容。

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。扩展元素对于一个非 SAML 定义的命名空间来说，必须是有效的命名空间。

— <RoleDescriptor>, <IDPSSODescriptor>, <SPSSODescriptor>, <AuthnAuthorityDescriptor>, <AttributeAuthorityDescriptor>, <PDPDescriptor> [一个或多个]; 或

— <AffiliationDescriptor> [必需的]

元素的基本内容是一个或多个角色描述符元素的序列，或定义一个联盟的专用描述符。

— <Organization> [可选的]

可选的元素，标明对由元素描述的负责 SAML 实体的组织。

— <ContactPerson> [零或多个]

可选的元素序列，标识各种的联系人员。

— <AdditionalMetadataLocation> [零或多个]

可选的有效命名空间位置序列，SAML 实体的其他元数据存在于这些位置。它可以包含其他格式的，或其他非 SAML 建议书描述的元数据。

任意从非 SAML 定义的命名空间中引入的有效命名空间的属性也可以被包含。

当用做一个元数据实例的根元素时，该元素必须包含一个 validUntil 或 cacheDuration 属性。建议仅元数据实例的根元素包含其中任一属性。

如果出现同一个类型的多个角色描述符元素，建议它们不要共享交迭的 protocolSupportEnumeration 值。对于在共享了一个 protocolSupportEnumeration 值的同一类型的多个角色描述符元素中进行选择，本建议书没有定义，但可以由元数据协议子集进行定义，可能通过使用其他的具有区分性扩展属性来实现。

下面的 Schema 段定义了<EntityDescriptor>元素及其 **EntityDescriptorType** 复杂类型：

```
<element name="EntityDescriptor" type="md:EntityDescriptorType"/>
<complexType name="EntityDescriptorType">
       <sequence>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="md:Extensions" minOccurs="0"/>
              <choice>
                     <choice maxOccurs="unbounded">
                            <element ref="md:RoleDescriptor"/>
                            <element ref="md:IDPSSODescriptor"/>
                            <element ref="md:SPSSODescriptor"/>
                            <element ref="md:AuthnAuthorityDescriptor"/>
                            <element ref="md:AttributeAuthorityDescriptor"/>
                            <element ref="md:PDPDescriptor"/>
                     </choice>
                     <element ref="md:AffiliationDescriptor"/>
              </choice>
              <element ref="md:Organization" minOccurs="0"/>
              <element ref="md:ContactPerson" minOccurs="0"
maxOccurs="unbounded"/>
              <element ref="md:AdditionalMetadataLocation" minOccurs="0"
maxOccurs="unbounded"/>
       </sequence>
       <attribute name="entityID" type="md:entityIDType" use="required"/>
       <attribute name="validUntil" type="dateTime" use="optional"/>
       <attribute name="cacheDuration" type="duration" use="optional"/>
       <attribute name="ID" type="ID" use="optional"/>
       <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
```

9.1.3.2.1 元素<Organization>

<Organization>元素规定有关负责 SAML 实体或角色负责的组织的基本信息。该元素的使用常常是可选的。其内容在本质上是资料性的，不直接映射到任何核心 SAML 元素或属性的。它的 **OrganizationType** 复杂类型由以下元素组成：

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。该元素的扩展禁止一定不包含全局的（命名空间无效的）元素或经由该元素中 SAML 定义的命名空间确认有效的元素。

— <OrganizationName> [一个或多个]

一个或多个符合语言规定的名称，可能或可能不适合人类消费。

— <OrganizationDisplayName> [一个或多个]

一个或多个符合语言规定的名称，适合人类消费。

— <OrganizationURL> [一个或多个]

一个或多个符合语言规定的 URI，指明指导用户获得额外信息的位置。语言合格性涉及特定位置的材料内容。

任意从非 SAML 定义的命名空间中引入的有效命名空间的属性也可以被包含。

下面的 Schema 段定义了<Organization>元素及其 **OrganizationType** 复杂类型：

```
<element name="Organization" type="md:OrganizationType"/>
<complexType name="OrganizationType">
      <sequence>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:OrganizationName" maxOccurs="unbounded"/>
            <element ref="md:OrganizationDisplayName"
maxOccurs="unbounded"/>
            <element ref="md:OrganizationURL" maxOccurs="unbounded"/>
      </sequence>
      <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
<element name="OrganizationName" type="md:localizedNameType"/>
<element name="OrganizationDisplayName" type="md:localizedNameType"/>
<element name="OrganizationURL" type="md:localizedURIType"/>
```

9.1.3.2.2 元素<ContactPerson>

<ContactPerson>元素详细说明对 SAML 实体或角色负一定责任的个人的基本联系信息。该元素的使用常常是可选的。其内容在本质上是资料性的，不直接映射到任何 SAML 元素或属性的核心。它的 **ContactType** 复杂类型由以下元素和属性组成：

— contactType [必需的]

指明使用 **ContactTypeType** 枚举的联系的类型。其可能值是 technical，support，administrative，billing 和 other。

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。扩展元素对于一个非 SAML 定义的命名空间来说，必须是有效的命名空间。

— <Company> [可选的]

可选字符串元素，指明联系人的公司名称。

— <GivenName> [可选的]

可选字符串元素，指明联系人的名字（第一个）。

— <SurName> [可选的]

可选字符串元素，指明联系人的别名。

— <EmailAddress> [零或多个]

零或多个包含 mailto: URI 代表属于联系人的 e-mail 地址。

— <TelephoneNumber> [零或多个]

零或多个字符串元素，指明联系人的电话号码。

任意从非 SAML 定义的命名空间中引入的有效命名空间的属性也可以被包含。

下面的 Schema 段定义了<ContactPerson>元素及其 **ContactType** 复杂类型：

```
<element name="ContactPerson" type="md:ContactType"/>
<complexType name="ContactType">
      <sequence>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:Company" minOccurs="0"/>
            <element ref="md:GivenName" minOccurs="0"/>
            <element ref="md:SurName" minOccurs="0"/>
            <element ref="md:EmailAddress" minOccurs="0"
maxOccurs="unbounded"/>
            <element ref="md:TelephoneNumber" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
      <attribute name="contactType" type="md:ContactTypeType"
use="required"/>
      <anyAttribute namespace="##other" processContents="lax"/>
```

```
</complexType>
<element name="Company" type="string"/>
<element name="GivenName" type="string"/>
<element name="SurName" type="string"/>
<element name="EmailAddress" type="anyURI"/>
<element name="TelephoneNumber" type="string"/>
<simpleType name="ContactTypeType">
       <restriction base="string">
              <enumeration value="technical"/>
              <enumeration value="support"/>
              <enumeration value="administrative"/>
              <enumeration value="billing"/>
              <enumeration value="other"/>
       </restriction>
</simpleType>
```

9.1.3.2.3 元素<AdditionalMetadataLocation>

<AdditionalMetadataLocation>元素是一个命名空间有效的 URI，它指明 SAML 实体的其他基于 XML 的元数据可以存在的位置。其 **AdditionalMetadataLocationType** 复杂类型使用一个命名空间属性（其类型同样是 **anyURI**）对 **anyURI** 类型进行扩展。这个必需的属性必须包含在特定位置建立的实例文档的根元素的 XML 命名空间。

下面的 Schema 段定义了<AdditionalMetadataLocation>元素及其 **AdditionalMetadataLocationType** 复杂类型：

```
<element name="AdditionalMetadataLocation"
type="md:AdditionalMetadataLocationType"/>
<complexType name="AdditionalMetadataLocationType">
       <simpleContent>
              <extension base="anyURI">
                     <attribute name="namespace" type="anyURI"
use="required"/>
              </extension>
       </simpleContent>
</complexType>
```

9.1.4 角色描述符元素

本节中的元素构成了绝大多数的元数据操作性支持元素。每个元素（除抽象元素外）定义了一个特定的支持 SAML 协议子集的操作性行为集合。

9.1.4.1 元素<RoleDescriptor>

<RoleDescriptor>元素是一个抽象的扩展点，它包含普通的描述性信息，目的是在不同的角色之间提供处理的通用性。新的角色可以通过扩展其抽象的 **RoleDescriptorType** 复杂类型来定义，该类型包含以下元素和属性：

— ID [可选的]

元素档案的唯一性标识符，其典型应用是在签字时作为一个参考点。

— validUntil [可选的]

可选属性，指出包含在元素和任意被包含的元素中的元数据的废止时间。

— cacheDuration [可选的]

可选属性，指出消费方应该缓存包含在元素和任意被包含的元素中的元数据的最大时间长度。

— protocolSupportEnumeration [必需的]

一组以空白为界的 URI，标识角色元素所支持的协议规范簇。对于 SAML V2.0 实体来说，该组必须包含 SAML 协议命名空间 URI，urn:oasis:names:tc:SAML:2.0:protocol。后续的 SAML 建议可以共享同一个命名空间 URI，但应该提供预备的“协议支持”标识符，以便在需要时能确保区分性。

— errorURL [可选的]

可选的 URI 属性，指定一个位置来指导用户解析问题和获得与此角色相关的其他支持。

— <ds:Signature> [可选的]

XML 签字，用来鉴别包含的元素及其内容。

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。扩展元素对于一个非 SAML 定义的命名空间来说，必须是有效的命名空间。

— <KeyDescriptor> [零或多个]

可选的元素序列，提供实体在角色中活动时所使用的密钥信息。

— <Organization> [可选的]

可选元素，指定与此角色相关联的组织。与<EntityDescriptor>元素中使用的元素等同。

— <ContactPerson> [零或多个]

可选的元素序列，标明与此角色相关联的联系。与<EntityDescriptor>元素中使用的元素等同。

任意从非 SAML 定义的命名空间中引入的有效命名空间的属性也可以被包含。

下面的 Schema 段定义了<RoleDescriptor>元素及其 **RoleDescriptorType** 复杂类型：

```
<element name="RoleDescriptor" type="md:RoleDescriptorType"/>
<complexType name="RoleDescriptorType" abstract="true">
       <sequence>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="md:Extensions" minOccurs="0"/>
              <element ref="md:KeyDescriptor" minOccurs="0"
maxOccurs="unbounded"/>
              <element ref="md:Organization" minOccurs="0"/>
              <element ref="md:ContactPerson" minOccurs="0"
maxOccurs="unbounded"/>
       </sequence>
       <attribute name="ID" type="ID" use="optional"/>
       <attribute name="validUntil" type="dateTime" use="optional"/>
       <attribute name="cacheDuration" type="duration" use="optional"/>
       <attribute name="protocolSupportEnumeration" type="md:anyURIListType"
use="required"/>
       <attribute name="errorURL" type="anyURI" use="optional"/>
       <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
<simpleType name="anyURIListType">
       <list itemType="anyURI"/>
</simpleType>
```

9.1.4.1.1 元素<KeyDescriptor>

<KeyDescriptor>元素提供关于实体用来签署数据或接收加密的密钥信息，和其他的加密细节。其 **KeyDescriptorType** 复杂类型由以下的元素和属性组成：

— use [可选的]

可选属性，指明所描述的密钥的用途。取值于 **KeyTypes** 枚举，由取值的加密和签字组成。

— <ds:KeyInfo> [必需的]

可选元素，直接或间接的标识密钥。参见 W3C XML 签字以获得有关该元素使用的更多细节。

— <EncryptionMethod> [零或多个]

可选元素，指明实体支持的算法和特定算法集。其准确内容根据支持的算法而变化。参见 W3C 加密以获得该元素的 **xenc:EncryptionMethodType** 复杂类型的定义。

下面的 Schema 段定义了<KeyDescriptor>元素及其 **KeyDescriptorType** 复杂类型：

```
<element name="KeyDescriptor" type="md:KeyDescriptorType"/>
<complexType name="KeyDescriptorType">
    <sequence>
        <element ref="ds:KeyInfo"/>
        <element ref="md:EncryptionMethod" minOccurs="0"
maxOccurs="unbounded"/>
    </sequence>
    <attribute name="use" type="md:KeyTypes" use="optional"/>
</complexType>
<simpleType name="KeyTypes">
    <restriction base="string">
        <enumeration value="encryption"/>
        <enumeration value="signing"/>
    </restriction>
</simpleType>
<element name="EncryptionMethod" type="xenc:EncryptionMethodType"/>
```

9.1.4.2 复杂类型SSODescriptorType

SSODescriptorType 抽象类型是具体类型 **SPSSODescriptorType** 和 **IDPSSODescriptorType** 的公共基类，这两个具体类型在后续各节描述。它使用反映了支持 SSO 的身份提供者和服务提供商共有协议子集的元素来扩展 **RoleDescriptorType**，并包含以下的附加元素：

— <ArtifactResolutionService> [零或多个]

零或多个 **IndexedEndpointType** 类型元素，描述支持第 12 节中定义的 Artifact Resolution 协议子集的索引端点。ResponseLocation 属性必须省略。

— <SingleLogoutService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Single Logout 协议子集的端点。

— <ManageNameIDService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Name Identifier Management 协议子集的端点。

— <NameIDFormat> [零或多个]

零或多个 **anyURI** 类型元素，列举在此角色中活动的该系统实体所支持的名称标识符格式。

下面的 Schema 段定义了 **SSODescriptorType** 复杂类型：

```
<complexType name="SSODescriptorType" abstract="true">
      <complexContent>
            <extension base="md:RoleDescriptorType">
                  <sequence>
                        <element ref="md:ArtifactResolutionService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:SingleLogoutService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:ManageNameIDService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
<element name="ArtifactResolutionService" type="md:IndexedEndpointType"/>
<element name="SingleLogoutService" type="md:EndpointType"/>
```

```
<element name="ManageNameIDService" type="md:EndpointType"/>
<element name="NameIDFormat" type="anyURI"/>
```

9.1.4.3 元素<IDPSSODescriptor>

<IDPSSODescriptor>元素通过反映支持 SSO 的身份提供者的协议子集的内容来扩展 **SSODescriptorType**。其 **IDPSSODescriptorType** 复杂类型包含以下附加元素和属性：

— WantAuthnRequestsSigned [可选的]

可选属性，指出身份提供者所接收到的<samlp:AuthnRequest>消息是否有签字的需求。如果省略，则其值假定为假。

— <SingleSignOnService> [一或多个]

一或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Authentication Request 协议的协议子集的端点。根据定义，所有的身份提供者都支持至少一个这样的端点。ResponseLocation 属性必须省略。

— <NameIDMappingService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Name Identifier Mapping 协议子集的端点。ResponseLocation 属性必须省略。

— <AssertionIDRequestService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 10 节中定义的 Assertion Request 协议或对断言请求的专用 URI 绑定的协议子集的端点。

注 1（资料性的）—PE33（参见OASIS PE:2006）建议将Assertion Request协议换成Assertion Query/Request。

— <AttributeProfile> [零或多个]

零或多个 **anyURI** 类型元素，列举该身份提供者支持的属性协议子集。

— <saml:Attribute> [零或多个]

零或多个元素，标明身份提供者支持的 SAML 属性。可以选择性的包含一些特定值，以指出仅支持属性定义所允许的某几个值。在该关联中，对某个属性的“支持”意味着身份提供者在单独登录过程中递送断言时可以包含该属性。

注 2（资料性的）— PE7（参见OASIS PE:2006）建议在上段的末尾添加以下文字：

WantAuthnRequestsSigned 属性意在为服务提供商指出它们能否期望一条未签字的<AuthnRequest>消息被身份提供者所接受。身份提供者不负责拒绝未签字的请求，服务提供商也不承担对其请求签字的责任，尽管它可以合理地预期一条未签字的请求会被拒绝。在某些情况下，服务提供商甚至可以不知道哪个身份提供者会最终接收并响应其请求，因此在这种情况下该属性的使用不能被严格地定义。而且，要注意可以预期的特定签字方法是绑定相关的。第10.2.4节中的HTTP Redirect绑定要求签字应用在URL编码的值中，而不是放在XML消息中，而其他映射一般准许签字以通常方式置于消息中。

下面的 Schema 段定义了<IDPSSODescriptor>元素及其 **IDPSSODescriptorType** 复杂类型：

```
<element name="IDPSSODescriptor" type="md:IDPSSODescriptorType"/>
<complexType name="IDPSSODescriptorType">
      <complexContent>
            <extension base="md:SSODescriptorType">
                  <sequence>
                        <element ref="md:SingleSignOnService"
maxOccurs="unbounded"/>
                        <element ref="md:NameIDMappingService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:AssertionIDRequestService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:AttributeProfile" minOccurs="0"
maxOccurs="unbounded"/>
```

```
                        <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
                    </sequence>
                    <attribute name="WantAuthnRequestsSigned" type="boolean"
use="optional"/>
             </extension>
       </complexContent>
</complexType>
<element name="SingleSignOnService" type="md:EndpointType"/>
<element name="NameIDMappingService" type="md:EndpointType"/>
<element name="AssertionIDRequestService" type="md:EndpointType"/>
<element name="AttributeProfile" type="anyURI"/>
```

9.1.4.4 元素<SPSSODescriptor>

<SPSSODescriptor>元素通过反映针对服务提供商的协议子集的内容来扩展 **SSODescriptorType**。其 **SPSSODescriptorType** 复杂类型包含以下附加元素和属性：

— AuthnRequestsSigned [可选的]

可选属性，指出服务提供商所发送的<samlp:AuthnRequest>消息是否会被签字。如省略，其值假定为假。

注 1（资料性的）— PE7（参见OASIS PE:2006）建议在上段的末尾添加以下文字：

该值为假（或该属性省略）不意味着服务提供商始终不对其请求签字或者一个签字的请求将被视作错误。然而，如果一个身份242提供者从一个服务提供商那里接收到一个未签字的<samlp:AuthnRequest>消息，而该服务提供商的元数据中又包含着值为真的该属性，则必须返回一个SAML错误响应并禁止完成请求。

— WantAssertionsSigned [可选的]

可选属性，指出该服务提供商所接收到的<samlp: Assertion>元素是否有签字的需求。如果省略，则其值假定为假。该需求是附加于其他源自特定配置/映射组合的签字需求之上的一个需求。

注 2（资料性的）— PE7（参见OASIS PE:2006）建议在上段的末尾添加以下文字：

注意在SAML绑定或协议层的封装签字不够满足该需求，例如签署一个包含断言的<samlp:Response>或一个TLS连接。

— <AssertionConsumerService> [一或多个]

一或多个元素，描述支持本建议书中定义的 Authentication Request 协议的协议子集的索引端点。根据定义，所有的服务提供商都支持至少一个这样的端点。

— <AttributeConsumingService> [零或多个]

零或多个元素，描述需要或期望使用 SAML 属性的服务提供商提供的应用或服务。

至多一个<AttributeConsumingService>元素的属性缺省值可以设为真。允许所包含的元素都不带有值为真的缺省属性。

下面的 Schema 段定义了<SPSSODescriptor>元素及其 **SPSSODescriptorType** 复杂类型：

```
<element name="SPSSODescriptor" type="md:SPSSODescriptorType"/>
<complexType name="SPSSODescriptorType">
       <complexContent>
             <extension base="md:SSODescriptorType">
                    <sequence>
                          <element ref="md:AssertionConsumerService"
maxOccurs="unbounded"/>
                          <element ref="md:AttributeConsumingService"
minOccurs="0" maxOccurs="unbounded"/>
                    </sequence>
                    <attribute name="AuthnRequestsSigned" type="boolean"
use="optional"/>
                    <attribute name="WantAssertionsSigned" type="boolean"
use="optional"/>
```

```
            </extension>
      </complexContent>
</complexType>
<element name="AssertionConsumerService" type="md:IndexedEndpointType"/>
```

9.1.4.4.1 元素<AttributeConsumingService>

就服务所需或期望的属性而言，<AttributeConsumingService>元素定义了一个特殊的由服务提供商提供的服务。其 **AttributeConsumingServiceType** 复杂类型包含以下元素和属性：

— index [必需的]

一个必需的属性，分配一个唯一的整形值给元素，以便它能在一个协议消息中被引用。

— isDefault [可选的]

标识服务提供商支持的缺省服务。如果特定的服务没有以别的方式在应用环境中指明时是有用的。如果省略，该值假定为假。

— <ServiceName> [一或多个]

一或多个符合语言规定的服务名称。

— <ServiceDescription> [零或多个]

零或多个符合语言规定的描述服务的字符串。

— <RequestedAttribute> [一或多个]

一或多个元素，指定该服务需要的或期望的属性。

下面的 Schema 段定义了<AttributeConsumingService>元素及其 **AttributeConsumingServiceType** 复杂类型：

```
<element name="AttributeConsumingService"
type="md:AttributeConsumingServiceType"/>
<complexType name="AttributeConsumingServiceType">
      <sequence>
            <element ref="md:ServiceName" maxOccurs="unbounded"/>
            <element ref="md:ServiceDescription" minOccurs="0"
maxOccurs="unbounded"/>
            <element ref="md:RequestedAttribute" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="index" type="unsignedShort" use="required"/>
      <attribute name="isDefault" type="boolean" use="optional"/>
</complexType>
<element name="ServiceName" type="md:localizedNameType"/>
<element name="ServiceDescription" type="md:localizedNameType"/>
```

9.1.4.4.2 元素<RequestedAttribute>

<RequestedAttribute>元素指定了在一个特定 SAML 属性中服务提供商的重要性，可选的包含特定值。其 **RequestedAttributeType** 复杂类型通过以下属性扩展了 **saml:AttributeType**：

— isRequired [可选的]

可选的 XML 属性，指出服务究竟是否需要相应的 SAML 属性来运行（而不是仅仅发现一个属性是否有用或值得要）。

如果包含特定的<saml:AttributeValue>元素，那么只有相匹配的值才和服务相关。

下面的 Schema 段定义了<RequestedAttribute>元素及其 **RequestedAttributeType** 复杂类型：

```
<element name="RequestedAttribute" type="md:RequestedAttributeType"/>
<complexType name="RequestedAttributeType">
  <complexContent>
        <extension base="saml:AttributeType">
              <attribute name="isRequired" type="boolean"
use="optional"/>
        </extension>
  </complexContent>
</complexType>
```

9.1.4.5 元素<AuthnAuthorityDescriptor>

<AuthnAuthorityDescriptor>元素使用反映针对认证授权中心、响应<samlp:AuthnQuery>消息的 SAML 权威的协议子集的内容来扩展 **RoleDescriptorType**。其 **AuthnAuthorityDescriptorType** 复杂类型包含以下附加元素：

— <AuthnQueryService> [一或多个]

一或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Authentication Query 协议的协议子集的端点。根据定义，所有的认证授权中心都支持至少一个这样的端点。

— <AssertionIDRequestService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Assertion Request 协议或第 10 节中定义的对于断言请求专用 URI 绑定的协议子集的端点。

— <NameIDFormat> [零或多个]

零或多个 **anyURI** 类型元素，列举该权威所支持的名称标识符格式（参见 8.7.3 该元素的可能取值）。

下面的 Schema 段定义了<AuthnAuthorityDescriptor>元素及其 **AuthnAuthorityDescriptorType** 复杂类型：

```
<element name="AuthnAuthorityDescriptor"
type="md:AuthnAuthorityDescriptorType"/>
<complexType name="AuthnAuthorityDescriptorType">
      <complexContent>
            <extension base="md:RoleDescriptorType">
                  <sequence>
                        <element ref="md:AuthnQueryService"
maxOccurs="unbounded"/>
                        <element ref="md:AssertionIDRequestService"
minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                  </sequence>
            </extension>
      </complexContent>
</complexType>
<element name="AuthnQueryService" type="md:EndpointType"/>
```

9.1.4.6 元素<PDPDescriptor>

<PDPDescriptor>元素使用反映针对策略决定点、响应<samlp:AuthzDecisionQuery>消息的 SAML 权威的协议子集的内容来扩展 **RoleDescriptorType**。其 **PDPDescriptorType** 复杂类型包含以下附加元素：

— <AuthzService> [一或多个]

一或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Authorization Decision Query 协议的协议子集的端点。根据定义，所有的策略决定点都支持至少一个这样的端点。

— <AssertionIDRequestService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Assertion Request 协议或第 10 节中定义的对于断言请求专用 URI 绑定的协议子集的端点。

注（资料性的）— PE33（参见OASIS PE:2006）建议将Assertion Request协议换成Assertion Query/Request。

— <NameIDFormat> [零或多个]

零或多个 **anyURI** 类型元素，列举该权威所支持的名称标识符格式（参见第 8.7.3 节该元素的某些可能取值）。

下面的 Schema 段定义了<PDPDescriptor>元素及其 **PDPDescriptorType** 复杂类型：

```
<element name="PDPDescriptor" type="md:PDPDescriptorType"/>
<complexType name="PDPDescriptorType">
       <complexContent>
              <extension base="md:RoleDescriptorType">
                     <sequence>
                            <element ref="md:AuthzService"
maxOccurs="unbounded"/>
                            <element ref="md:AssertionIDRequestService"
minOccurs="0" maxOccurs="unbounded"/>
                            <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
<element name="AuthzService" type="md:EndpointType"/>
```

9.1.4.7 元素<AttributeAuthorityDescriptor>

<AttributeAuthorityDescriptor> 元素使用反映针对属性授权中心、响应 <samlp:AttributeQuery> 消息的 SAML 权威的协议子集的内容来扩展 **RoleDescriptorType**。其 **AttributeAuthorityDescriptorType** 复杂类型包含以下附加元素：

— <AttributeService> [一或多个]

一或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Attribute Query 协议的协议子集的端点。根据定义，所有的属性授权中心都支持至少一个这样的端点。

— <AssertionIDRequestService> [零或多个]

零或多个 **EndpointType** 类型元素，描述支持第 12 节中定义的 Assertion Request 协议或第 10 节中定义的对于断言请求专用 URI 绑定的协议子集的端点。

注（资料性的）— PE33（参见OASIS PE:2006）建议将Assertion Request协议换成Assertion Query/Request。

— <NameIDFormat> [零或多个]

零或多个 **anyURI** 类型元素，列举该权威所支持的名称标识符格式（参见第 8.7.3 节该元素的可能取值）。

— <AttributeProfile> [零或多个]

零或多个 **anyURI** 类型元素，列举该权威所支持的属性协议子集（参见第 8.7.3 节该元素的可能取值）。

— <saml:Attribute> [零或多个]

零或多个元素，标明权威所支持的 SAML 属性。可以选择性地包含一些特定值，以指出仅支持属性定义所允许的某几个值。

下面的 Schema 段定义了 <AttributeAuthorityDescriptor> 元素及其 **AttributeAuthorityDescriptorType** 复杂类型：

```
<element name="AttributeAuthorityDescriptor"
type="md:AttributeAuthorityDescriptorType"/>
<complexType name="AttributeAuthorityDescriptorType">
       <complexContent>
              <extension base="md:RoleDescriptorType">
                     <sequence>
                            <element ref="md:AttributeService"
maxOccurs="unbounded"/>
                            <element ref="md:AssertionIDRequestService"
minOccurs="0" maxOccurs="unbounded"/>
                            <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                            <element ref="md:AttributeProfile" minOccurs="0"
maxOccurs="unbounded"/>
                            <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
                     </sequence>
              </extension>
       </complexContent>
</complexType>
<element name="AttributeService" type="md:EndpointType"/>
```

9.1.5 元素<AffiliationDescriptor>

当<EntityDescriptor>描述的是 SAML 实体的联盟（典型的有服务提供商）而不是单个实体时，<AffiliationDescriptor>元素可以作为一组角色描述符序列的替代。<AffiliationDescriptor>元素提供组成联盟的各个实体的概要信息和联盟自身的一般信息。其 **AffiliationDescriptorType** 复杂类型包含以下元素和属性：

— affiliationOwnerID [必需的]

指定对负责联盟的实体的唯一标识符。其所有者不一定是联盟的一个成员；如果是，则其标识符必须也出现在一个<AffiliateMember>元素中。

— ID [可选的]

是元素的一个档案唯一性的标识符，其典型应用是在签字时作为一个参考点。

— validUntil [可选的]

可选属性，指出包含在元素和任意被包含的元素中的元数据的废止时间。

— cacheDuration [可选的]

可选属性，指出消费方应该缓存包含在元素和任意被包含的元素中的元数据的最大时间长度。

— <ds:Signature> [可选的]

一个 XML 签字，来鉴别包含的元素及其内容（参见第 8 节）。

— <Extensions> [可选的]

它包含可选的元数据扩展，这些扩展在元数据发行方和消费方之间已达成一致。扩展元素对于一个非 SAML 定义的命名空间来说，必须是有效的命名空间。

— <AffiliateMember> [一或多个]

一或多个元素，通过指定每个成员的唯一标识符来列举联盟的成员（同时参见第 8.7.3.6 节）。

— <KeyDescriptor> [零或多个]

可选的元素序列，提供关于联盟统一使用的密钥信息，它不同于联盟中的个体成员使用的，在这些实体的元数据中公布的密钥。

任意从非 SAML 定义的命名空间中引入的有效命名空间的属性也可以被包含。

下面的 Schema 段定义了<AffiliationDescriptor>元素及其 **AffiliationDescriptorType** 复杂类型：

```
<element name="AffiliationDescriptor" type="md:AffiliationDescriptorType"/>
<complexType name="AffiliationDescriptorType">
       <sequence>
              <element ref="ds:Signature" minOccurs="0"/>
              <element ref="md:Extensions" minOccurs="0"/>
              <element ref="md:AffiliateMember" maxOccurs="unbounded"/>
              <element ref="md:KeyDescriptor" minOccurs="0"
maxOccurs="unbounded"/>
       </sequence>
       <attribute name="affiliationOwnerID" type="md:entityIDType"
use="required"/>
       <attribute name="validUntil" type="dateTime" use="optional"/>
       <attribute name="cacheDuration" type="duration" use="optional"/>
       <attribute name="ID" type="ID" use="optional"/>
       <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
<element name="AffiliateMember" type="md:entityIDType"/>
```

9.1.6 例子

下面是 SAML 系统实体作为身份提供者和属性授权中心的元数据举例。其中签字以占位符显示，不具有实际内容。

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
       xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
       xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
       entityID="https://IdentityProvider.com/SAML">
       <ds:Signature>...</ds:Signature>
    <IDPSSODescriptor WantAuthnRequestsSigned="true"

       protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
         <KeyDescriptor use="signing">
             <ds:KeyInfo>
                 <ds:KeyName>IdentityProvider.com SSO Key</ds:KeyName>
             </ds:KeyInfo>
         </KeyDescriptor>
         <ArtifactResolutionService isDefault="true" index="0"
                     Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                     Location="https://IdentityProvider.com/SAML/Artifact"/>
         <SingleLogoutService
                     Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                     Location="https://IdentityProvider.com/SAML/SLO/SOAP"/>
         <SingleLogoutService

       Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"

       Location="https://IdentityProvider.com/SAML/SLO/Browser"

       ResponseLocation="https://IdentityProvider.com/SAML/SLO/
Response"/>
         <NameIDFormat>
              urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName
         </NameIDFormat>
         <NameIDFormat>
              urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
         </NameIDFormat>
         <NameIDFormat>
              urn:oasis:names:tc:SAML:2.0:nameid-format:transient
         </NameIDFormat>
         <SingleSignOnService

       Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"

       Location="https://IdentityProvider.com/SAML/SSO/Browser"/>
         <SingleSignOnService

       Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
```

```
      Location="https://IdentityProvider.com/SAML/SSO/Browser"/>
       <saml:Attribute

      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
                   Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
                   FriendlyName="eduPersonPrincipalName">
       </saml:Attribute>
       <saml:Attribute

      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
                   Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"
                   FriendlyName="eduPersonAffiliation">
           <saml:AttributeValue>member</saml:AttributeValue>
           <saml:AttributeValue>student</saml:AttributeValue>
           <saml:AttributeValue>faculty</saml:AttributeValue>
           <saml:AttributeValue>employee</saml:AttributeValue>
           <saml:AttributeValue>staff</saml:AttributeValue>
       </saml:Attribute>
    </IDPSSODescriptor>
    <AttributeAuthorityDescriptor

      protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
       <KeyDescriptor use="signing">
           <ds:KeyInfo>
               <ds:KeyName>IdentityProvider.com AA Key</ds:KeyName>
           </ds:KeyInfo>
       </KeyDescriptor>
       <AttributeService
                   Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                   Location="https://IdentityProvider.com/SAML/AA/SOAP"/>
       <AssertionIDRequestService
                   Binding="urn:oasis:names:tc:SAML:2.0:bindings:URI"
                   Location="https://IdentityProvider.com/SAML/AA/URI"/>
       <NameIDFormat>
            urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName
       </NameIDFormat>
       <NameIDFormat>
            urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
       </NameIDFormat>
       <NameIDFormat>
            urn:oasis:names:tc:SAML:2.0:nameid-format:transient
       </NameIDFormat>
       <saml:Attribute

      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
                   Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
                   FriendlyName="eduPersonPrincipalName">
       </saml:Attribute>
       <saml:Attribute

      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
                   Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"
                   FriendlyName="eduPersonAffiliation">
           <saml:AttributeValue>member</saml:AttributeValue>
           <saml:AttributeValue>student</saml:AttributeValue>
           <saml:AttributeValue>faculty</saml:AttributeValue>
           <saml:AttributeValue>employee</saml:AttributeValue>
           <saml:AttributeValue>staff</saml:AttributeValue>
       </saml:Attribute>
    </AttributeAuthorityDescriptor>
    <Organization>
        <OrganizationName xml:lang="en">Identity Providers R
US</OrganizationName>
        <OrganizationDisplayName xml:lang="en">
             Identity Providers R US, a Division of Lerxst Corp.
        </OrganizationDisplayName>
        <OrganizationURL
xml:lang="en">https://IdentityProvider.com</OrganizationURL>
    </Organization>
</EntityDescriptor>
```

下面是 SAML 系统实体作为服务提供商的元数据举例。其中签字以占位符显示，不具有实际内容。为达到举例的目的，业务不要求用户具有唯一标识他们自身的身份，而是基于一个角色-行式属性授权准入。

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
      entityID="https://ServiceProvider.com/SAML">
      <ds:Signature>...</ds:Signature>
    <SPSSODescriptor AuthnRequestsSigned="true"

      protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
       <KeyDescriptor use="signing">
           <ds:KeyInfo>
               <ds:KeyName>ServiceProvider.com SSO Key</ds:KeyName>
           </ds:KeyInfo>
       </KeyDescriptor>
       <KeyDescriptor use="encryption">
           <ds:KeyInfo>
               <ds:KeyName>ServiceProvider.com Encrypt Key</ds:KeyName>
           </ds:KeyInfo>
           <EncryptionMethod
      Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
       </KeyDescriptor>
       <SingleLogoutService
                   Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                   Location="https://ServiceProvider.com/SAML/SLO/SOAP"/>
       <SingleLogoutService

      Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
                   Location="https://ServiceProvider.com/SAML/SLO/Browser"

      ResponseLocation="https://ServiceProvider.com/SAML/SLO/Response"/>
       <NameIDFormat>
            urn:oasis:names:tc:SAML:2.0:nameid-format:transient
       </NameIDFormat>
       <AssertionConsumerService isDefault="true" index="0"

      Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact"

      Location="https://ServiceProvider.com/SAML/SSO/Artifact"/>
       <AssertionConsumerService index="1"

      Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                   Location="https://ServiceProvider.com/SAML/SSO/POST"/>
       <AttributeConsumingService index="0">
           <ServiceName xml:lang="en">Academic Journals R US</ServiceName>
           <RequestedAttribute

      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-
format:uri"
                           Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"
                           FriendlyName="eduPersonEntitlement">
               <saml:AttributeValue>
                   https://ServiceProvider.com/entitlements/123456789
               </saml:AttributeValue>
           </RequestedAttribute>
       </AttributeConsumingService>
    </SPSSODescriptor>
    <Organization>
        <OrganizationName xml:lang="en">Academic Journals R
US</OrganizationName>
        <OrganizationDisplayName xml:lang="en">
              Academic Journals R US, a Division of Dirk Corp.
        </OrganizationDisplayName>
        <OrganizationURL
xml:lang="en">https://ServiceProvider.com</OrganizationURL>
    </Organization>
</EntityDescriptor>
```

9.2 签字处理

元数据实例中的各种元素可以被数字签字（如<ds:Signature>元素的元素包含所指明的）。数字签字具有以下好处：

9.2.1 元数据完整性

元数据由一个可信任的签字人认证。

并不总是需要数字签字，例如，如果信任方直接通过一种安全通道（没有中间物），从发布实体那里直接获得信息，而这个实体已经通过非数字签字的方法通过信任方的认证的话，那么并不总是需要数字签字。

“直接”认证和两个团体间安全通道的建立可以有很多不同的技术。包括 TLS、HMAC 和基于密码的机制等等。此外，可应用的安全需求依赖于通信应用。

另外，元素还可以继承封装父元素的签字，这些父元素由自己签字。

当这样的关联不出现时，建议至少元数据实例的根元素应被签字。

9.2.2 XML签字协议子集

W3C XML 签字规范为签字数据产生一个具有弹性和多种选择的通用 XML 句法。本节详细说明这些对这些设施的约束性，这样元数据处理器不必处理 XML 签字处理的所有一般性。这种用法特别使用了 **xs:ID**-typed 属性，该属性可选地出现在应用签字的元素上。这些属性作为身份标识符的标属性在本节全部有所涉及。

1） 签字格式和算法

XML签字有三种方法来关联文件和签字：封装、已封装和分离。

SAML元数据在为本建议书中定义的元素签字时必须使用已封装的签字。SAML处理器应该支持RSA签字和公钥操作认证，算法应符合在http://www.w3.org/2000/09/xmldsig#rsa-sha1 中的定义。

2） 引用

已签字的元数据元素必须为已签字元素的身份属性提供值。元素标识的算法可以是也可以不是包含已签字元数据元素的实际XML文件的根元素。

签字必须包含一个<ds:Reference>，<ds:Reference>包含了正在被签字的元数据元素的身份属性值的URI引用。例如，如果标识符属性值是“foo”，那么<ds:Reference>中的URI属性必须是"#foo"。

因而一个元数据元素的签字必须应用到已签字元素和它包含的所有子元素的内容上。

3） 规范方法

SAML实现应该使用有或者没有注释的Exclusive Canonicalization，作为<ds:Transform>算法，二者都在<ds:SignedInfo>的<ds:CanonicalizationMethod>元素中。使用Exclusive Canonicalization能够保证创建在SAML元数据上的签字的验证可以独立于XML关联进行，这些元数据是封装在XML关联中的。

4） 转换

SAML元数据中的签字不应包含变换而是包含被封装签字转换（标识符http://www.w3.org/2000/09/xmldsig# enveloped-signature）或独立规范转换（标识符http://www.w3.org/2001/10/xml-exc-c14n# 或http://www.w3.org/2001/10/xml-exc-c14n#WithComments）之外，不必包含转换。

签字的验证者可能会当做非法来拒绝包含其他转换算法的签字。如果验证者没有拒绝，它必须保证SAML消息的全部内容都被签字。这种保证可以通过建立那些可接受转换的带外协议，或者通过手动转换内容，并对相同SAML消息组成的转换结果进行再次验证来实现。

5） **KeyInfo**

W3C XML签字定义了<ds:KeyInfo>元素的使用。SAML不需要使用<ds:KeyInfo>元素，也不用对它的使用强加任何限制。因此，<ds:KeyInfo>元素可以不出现。

9.3 元数据的发布和解析

在本建议书中，为实体发布元数据文件（为用户解决文件的位置）提供了两种机制：通过“众所周知的位置”直接公布实体的唯一标识符（称作 *entityID* 或 *providerID* 的 URI），或间接地在 DNS 中发布元数据的位置。当然也允许其他的带外机制。支持这两种方法的用户必须在使用“众所周知的位置”机制之前使用 DNS 来尝试解析。

当检索需要文件进行网络传输时，传输应采用提供服务器认证和完整性保护的机制来保护。例如，基于 HTTP 解析应当采用在 IETF RFC 2246 中定义的 TLS 来保护，RFC 2246 的修订版为 RFC 3546。

为了帮助元数据建立精确合法的信任，本节描述了不同的机制，包括使用 XML 签字，TLS 服务器认证，DNS 签字。不管使用哪种机制，信任方都应该在信任之前通过某种方法建立元数据信息的信任。

9.3.1 通过众所周知的位置发布和解析

下面小节描述了元数据通过众所周知的位置发布和解析。

9.3.1.1 发布

实体可能会通过在它的唯一标识符所指示的位置发布文档，而在众所周知的位置发布它们的元数据文档。唯一标识符必须采用 URL（而不是 URN）格式，强烈建议为此目的，使用 `https` URL 格式。如果文档没有直接放置在位置上，那么可能会使用 URL 方案（如 HTTP 1.1 302 重定向）所支持的间接机制。如果发布的协议允许基于 MIME 的内容类型鉴定，元数据实例的内容类型则必须是 `application/samlmetadata+xml`。

众所周知的位置提供的 XML 文档必须仅为那些由唯一标识符代表的实体（也就是，根元素必须是匹配位置的有着 `entityID` 的<EntityDescriptor>元素）描述元数据。如果有其他实体需要描述，那么必须使用<AdditionalMetadataLocation>元素。因为很多的实体不是按照这种标识符定义的，所以禁止利用这种机制在发布的文档中使用<EntitiesDescriptor>元素。

9.3.1.2 解析

如果实体的唯一的标识符是一个 URL，那么元数据用户可能会通过丢弃这个标识符，以一种专用方案的方法来直接解析实体的唯一标识符。

9.3.2 通过DNS发布和解析

为了更容易获得元数据文档，并为实体的唯一标识符和元数据位置之间提供附加的间接发布方式，实体可能会在 IETF RFC1034 中定义的相应 DNS 域中发布它们的元数据文档位置。实体的唯一标识符（URI）用做过程的输入。由于 URI 是一种灵活的标识符，所以位置的发布方法和解析过程都由 URI 方案和 fully-qualified 名称来决定。元数据的 URI 位置可以通过随后询问 IETF RFC 2914 和 IETF RFC 3403 中定义的 NAPTR 源记录（RR）来取得。

建议实体使用 IETF RFC 2535 中的已签字域文件来发布它们的源记录，这样信任方可以确定已发布位置的有效性、域的权威性和 DNS 响应的完整性。如果 DNS 域签字出现，那么信任方必须让这个签字完全有效。

9.3.2.1 发布

本建议书使用 IETF RFC 2915 和 IETF RFC 3403 中描述的 NAPTR 源记录。鼓励去熟悉这些文档。

动态授权发现系统（DDDS）是一个通用系统，它用来基于应用特定输入字符串和众所周知的规则的应用检索信息，对该字符串进行改变，直到达到中止条件，即要求在该特定应用定义的数据库中进行查找，或基于应用定义的规则对 URL 进行解析。DDDS 定义了 DNS Resource Record 的特殊类型，NAPTR 记录，用来存储使用 DDDS 规则所需要的 DNS 信息。

当需要分发多个元数据文件时，或因为要求独立的密钥材料的多种信任关系，要求不同的元数据文件时，或当服务接口要求独立的元数据断言时，实体可以发布独立的 URL。这可以通过使用可选的<AdditionalMetadataLocation>元素完成，或通过 regexp 工具和 NAPTR 资源记录自身中的不同服务定义域来实现。

如果发布协议允许基于 MIME 的内容类型鉴定，则元数据实例的内容类型必须是 application/samlmetadata+xml。

如果实体的唯一标识符是一个 URN，相应元数据位置的发布依照 IETF RFC 3404 中的规定进行。否则，元数据位置的解析按以下规定进行。

以下是 SAML 元数据解析的 DDDS 的特定应用协议子集：

1） 第一个众所周知的规则

对于SAML元数据解析处理的“第一个公众规则”是分析实体的唯一标识符并获取完全合格的域名（子式3）。

2） 次序字段

次序字段指出处理每一个返回的NAPTR资源记录的顺序。发布方可以提供多个NAPTR资源记录，它们必须按照该域指出的顺序由解析应用处理。

3） 优先字段

为了最终的NAPTR资源记录，发布方向解析应用表达使用的首选顺序。在业务字段的值与解析者的要求不符时，解析应用可以忽略该顺序（例如：资源记录返回一个应用不支持的协议）。

4） 标记字段

SAML元数据解析两次使用终结符“U”标记，以及Null值（意味着还有其他的资源记录待处理）。“U”标记说明规则的输出是一个URI。

5） 服务字段

在以下BNF中描述的特定的SAML服务字段，断言了使得实例文件可用的模式：

```
servicefield = 1("PID2U" / "NID2U") "+" proto [*(":" class) *(":" servicetype)]
proto = 1("https" / "uddi")
class = 1[ "entity" / "entitygroup" )
servicetype = 1(si / "spsso" / "idpsso" / "authn" / "authnauth" / "pdp" /
"attrauth" / alphanum )
si = "si" [":" alphanum] [":endpoint"]
alphanum = 1*32(ALPHA / DIGIT)
```

这里：

— servicefield PID2U解析一个实体的唯一标识符为元数据URL。

— servicefield NID2U解析一个负责人的<NameID>成为一个元数据URL。

— Proto描述检索复协议（https或uddi）。在UDDI的情况下，URL是一个引用WSDL文件的http(s) URL。

— Class识别引用元数据文件是描述单个实体还是多个实体。在后一种情况下，引用文件必须包含起始的唯一标识符定义的实体作为文件自身中的一组实体的成员，例如<AffiliationDescriptor>或<EntitiesDescriptor>。

— Servicetype允许实体为不同的角色和服务作为独立文件发布元数据。遇到多个servicetype断言的解析者会废弃合适的URI，依赖于操作所需的服务（例如：一个既作为身份提供者又作为服务提供商的实体能在不同位置为每个角色发布元数据）。Authn服务类型表征了一个<SingleSignOnService>端点。

— si（带有可选的端点元素）允许发布方直接为一个服务实例发布元数据，或通过连接一个SOAP端点（使用的端点）。

举例来说：

— PID2U+https:entity – 表示实体的全部元数据文件可通过https协议获得。

— PID2U+uddi:entity:si:foo – 表示描述服务实例“foo”的WSDL文件位置。

— PID2U+https:entitygroup:idpsso – 表示一组担当SSO身份提供者实体的元数据，起始的实体是其中的一个成员。

— NID2U+https:idp – 表示负责方的SSO身份提供者的元数据。

6） regex和置换字段

输入字符串通过regex处理后的预期输出必须是一个有效的https URL或UDDI节点（WSDL文件）地址。

9.3.2.2 NAPTR举例

本节列举出一些实体能够使用的、支持 NAPTR（参见 IETF RFC 2915）的 URL 和 e-mail 例子。

a） 实体元数据 NAPTR 例子

实体以下面的方式发布元数据 URL：

```
$ORIGIN provider.biz

;; order pref f service regexp or replacement

IN NAPTR 100 10 "U" PID2U+https:entity
 "!^.*$!https://host.provider.biz/some/directory/trust.xml!" ""
IN NAPTR 110 10 "U" PID2U+https: entity:trust
 "!^.*!https://foo.provider.biz:1443/mdtrust.xml!" ""
IN NAPTR 125 10 "U" PID2U+https:"
IN NAPTR 110 10 "U" PID2U+uddi:entity
 "!^.*$!https://this.uddi.node.provider.biz/libmd.wsdl" ""
```

b） 名称标识符例子

责任人的雇主 example.int 运作一个身份提供者，它可以被一个职责供应公司用来认证授权的买家。提供者采用用户的 email 地址 buyer@example.int 作为解析处理的输入，并分析 email 地址来得到 FQDN （example.int）。雇主在 example.int DNS 中发布以下的 NAPTR 记录：

```
$ORIGIN example.int

IN NAPTR 100 10 "U" NID2U+https:authn
 "!^([^@]+)@(.*)$!https://serv.example.int:8000/cgi-bin/getmd?\1!" ""
IN NAPTR 100 10 "U" NID2U+https:idp
 "!^([^@]+)@(.*)$!https://auth.example.int/app/auth?\1" ""
```

9.3.2.3 解析

当通过 DNS 为一个实体解析元数据时，要使用实体的唯一标识符作为解析处理的初始输入，而不是作为实际位置，按以下步骤进行：

— 如果唯一标识符是一个URN，按照IETF RFC 3403中的定义继续解析步骤。

— 否则，分析标识符以获得完全合格的域名。

— 重复查询域的NAPTR资源记录的DNS，直到返回一个最终的资源记录。

— 先依据结果集合的服务字段，其次是规则字段，然后是优先域来确定使用哪一个资源记录。

— 根据应用的要求在所提供的位置获取文件。

为启动元数据信息的位置解析，在某些情况下有必要分解实体的唯一标识符（表示为一个 URI）成为一个或多个子元素。

当启动分解过程时，应该使用以下的规则表达式：

```
^([^:/?#]+:)?/*([^:/?#]*@)?(([^/?:#]*\.)*(([^/?#:\.]+)\.([^/?#:\.]+)))(:\d+)?([^
?#]*)(\?[^#]*)?(#.*)?$
     1            2           34             56             7              8       9
10         11
```

子式 3 的结果必须是一个完全合格的域名（FQDN），它将作为从该域检索元数据位置的根据。

一旦完成标识符的分析，应用就会执行一个 DNS 查询，以获得 NAPTR 资源记录的结果字段（子式 5）；它应该期望一或多个响应。应用可以将任何与当前请求操作无关的服务定义从结果集合中排除。

解析应用随后必须依照次序字段排序结果集合，并可以基于优先集合整理结果集合。不需要解析者遵循优先字段的规则。会反复对作为结果的 NAPTR 资源记录进行操作（基于顺序标记），直到达到最终的 NAPTR 资源记录。

结果是一个格式化的、绝对的 URL，它随后被用来检索元数据文件。

9.3.2.4 元数据位置缓存

位置缓存禁止超出 DNS 域的 TTL 限制，位置就是从该域中取得的。当域的 TTL 到期时，解析方必须取得一个元数据位置的新副本。

元数据文件的发布方在对元数据文件位置进行改变时，应当谨慎的考虑到域的 TTL。假定有这样的位置改变发生，发布方必须要么在旧的位置和新的位置都保留文件，直到确保所有相应的解析方都获得更新后的位置（例如：域的时间改为+ TTL），要么在旧的位置提供一个 HTTP Redirect 响应，指定新的位置。

9.3.3 元数据的后处理

下面的各节描述了元数据的后处理过程。

9.3.3.1 元数据实例缓存

文件缓存禁止超出主体元素的 `validUntil` 或 `cacheDuration` 属性限制。如果元数据元素具有父元素，且父元素包含缓存策略，则父元素优先。

为了适当地处理 `cacheDuration` 属性，消费方在文件检索时必须保留日期和时间。

当一个文件或元素到期时，消费方必须重新获得一个新的副本，这可能要求文件地址进行一次更新。消费方应当按照 RFC 2616,13 的规定处理文件缓存过程，并且可以从 HTTP 服务器那里请求得到最近修改的日期和时间。发布方应当按 IETF RFC 2616，第 10.3.5 节（304 Not Modified）中描述的那样确保缓存处理是可接受的。

9.3.3.2 HTTPS重定向处理

发布方可以按 IETF RFC 2616 中所定义的发布一个 HTTP Redirect (301 Moved Permanently, 302 or 307 Temporary Redirect)，用户代理必须遵循 Redirect 响应中规定的 URL。Redirects 应该和初始的请求采用同样的协议。

9.3.3.3 XML签字和一般信任处理过程

元数据处理过程为元数据自身和由这些元数据描述的实体的信任协商提供了几种机制：

— 信任源自DNS域的签字，元数据位置URL就是从那里决定的，这确保了元数据文件位置的正确性；

— 信任源自元数据文档自身的签字处理过程，这确保了XML文件的完整性；

— 信任源自元数据位置URL的TLS服务器认证，这确保了元数据发布方的身份。

元数据文件的后处理过程必须包含 XML 文件级的签字处理并且可以包含其他两个处理中的一个。特别地，信任方在解析和分析处理中可以选择信任任何一个被引用的权威。元数据的发布方必须使用一种文件完整性机制，并可以使用其他两种处理配置中的任何一种来建立对由执行策略管理的元数据文件的信任。必须遵循以下需要考虑的事项：

1） 签字的DNS域处理

如果出现，则应当按照IETF RFC 2535之描述，处理DNS区域签字的验证。

2） 签字的文件和片段的处理

已发布的元数据文件应当签字，如本建议书所述，或者通过给文件主体，或者通过其他的可信任方签发一个证书。发布方可以考虑将其他团体的签字作为一种信任手段。

当出现时，元数据消费方必须按照本建议书的描述验证元数据文件的签字。

3） 通过TLS检索元数据过程中的服务器认证处理

强烈建议发布方实现TLS URL；因此，消费方应当考虑到从TLS证书发行方产生的信任继承性。发布内容URL并不总是位于元数据文件主体域内，因此，消费方不应假定证书的主体是可疑实体，它可能是由其他的可信任方控制的。

对于缓存文件，该信任机制可能不可用，此类情况下，应该使用其他机制。

10 SAML绑定

本节规定了在通信协议及框架中，为了使用 SAML 断言和请求-响应消息，规定 SAML 协议绑定。

SAML 请求-响应消息交互到标准消息传递或通信协议的映射称为 SAML 协议绑定（或仅称为绑定）。例如 SAML 请求-响应消息交互到某个特定通信协议<FOO>的绑定就可称为一个对于 SAML 的<*FOO*>绑定，或一个 SAML <*FOO*> 绑定。

举例来说，一个 SAML SOAP 绑定描述了 SAML 请求和响应消息互换是如何映射到 SOAP 消息互换的。

本建议书的目的是规定一组精选的映射，对它们进行足够详细的描述以保证各个独立的 SAML 一致性软件实现在使用标准信息传递或通信协议的时候能够协同工作。

除非特别指出，否则一个映射应被理解为支持来自 **samlp:RequestAbstractType** 和 **samlp:StatusResponseType** 类型中的任一 SAML 协议消息的传输。进一步来说，当一个映射涉及“SAML 请求和响应”时，它应被理解为来自那些类型中的任一协议消息。

本建议书在文本中采用如下的印刷约定：<ns:Element>，XML 属性，数据类型，其他关键字。在某些情况下，尖括号被用来指示非终结符，而不是 XML 元素；其用法通过关联即可不言自明。

10.1 规定其他协议绑定的指南

本建议书定义了一组精选的协议绑定，而其他的有可能在将来开发。本节为那些想规定其他映射的第三方方提供指导方针。以下是对每个协议绑定都必须提出的问题对照列表：

— 规定三种标识信息：一个是唯一标识该协议绑定的URI，另一个是该映射提出者的邮箱或电子联系信息，还有一个是相对于之前定义的映射或配置，新的映射更新或废除的内容的参考。

— 描述包含在绑定中的各方之间的交互作用。各方所采用的应用程序的任何限制和在每个交互作用中涉及的协议都必须明确地指出。

— 确认每个交互作用中涉及的各方，包括有与之相关的各方的数量，以及是否会包含中间媒介。

— 规定每个交互作用中涉及的各方的认证方法，包括是否需要认证，以及可接受的认证类型。

— 标识对于消息完整性的支持水平，包括为确保消息完整性所采用的机制。

— 标识对于可信度的支持水平，包括是否有第三方可能看到SAML消息和断言的内容，映射是否需要可信度，以及为取得可信度所推荐采用的机制。

— 标识错误状态，包括各个参与方，尤其是那些接收并处理SAML断言或消息的参与方的错误状态。

— 标识安全性方面的考虑，包括风险分析和对策描述。

— 标识元数据方面的考虑，这样对某个特殊的通信协议中涉及的或在某个特别的配置中使用的映射的支持，可以高效的、互操作的方式被通告。

10.2 协议绑定

以下的各节定义了作为 SAML 标准的一部分所规定的协议绑定。

10.2.1 一般性考虑

以下的各节描述了所有为 SAML 规定的协议绑定的特性。

10.2.1.1 RelayState的使用

某些映射为了保留并传送状态信息，定义了“RelayState”机制。当在 SAML 协议的起始步骤，即传送请求消息时采用该机制，它会在随后选择用来传送响应的映射中设置请求。也就是说，如果一个 SAML 请求消息带有 RelayState 数据，那么 SAML 响应端必须采用一个同样支持 RelayState 机制的映射来返回 SAML 协议响应，并且必须将其在请求中接收到的准确的 RelayState 数据置于响应中相应的 RelayState 参数中。

10.2.1.2 安全性

除非特别说明，否则这些安全性断言适用于所有映射。映射也可以对这些安全特性做出附加声明。

1） TLS 1.0 的使用

除非特别规定，否则在任何的SAML绑定中使用TLS 1.0（IETF RFC 2246），服务器端必须使用X.509 v3证书对客户端认证。客户端必须基于该证书的内容验明服务器方的身份（典型做法是通过证书的subject DN字段，`subjectAltName`属性等进行审查）。

2） 数据起始认证

与某个消息相关联的SAML请求端和SAML响应端两者的认证都是可选的，依赖于使用环境。在SOAP消息交换层或从内在的底层协议有效认证机制（例如许多绑定TLS或HTTP协议）可以被用来提供数据起始认证。

在SAML协议消息需要通过中间媒介的情况下，传输认证将不能满足映射中端到端的起始认证要求 — 在此情况下建议使用消息认证。

SAML自身提供了各方相互认证的机制，除此以外，SAML可以使用其他认证机制为SAML自身提供安全保障。

3） 消息完整性

SAML请求端和SAML响应端两者的消息完整性都是可选的，并依赖于使用环境。内在的底层协议中的安全层或SOAP消息交换层采用的机制可以被用来确保消息完整性。

在SAML协议消息需要通过中间媒介的情况下，传输完整性将不能满足映射中端到端的完整性要求——在此情况下建议检查消息完整性。

4） 消息可信度

SAML请求端和SAML响应端两者的消息可信度都是可选的，并依赖于使用环境。内在的底层协议中的安全层或SOAP消息交换层采用的机制可以被用来确保消息可信度。

在SAML协议消息需要通过中间媒介的情况下，传输可信度将不能满足绑定中端到端的可信度要求。

5） 其他安全性考虑

认证、消息完整性和可信度机制的每种组合在配置之前，应该先针对特定协议交换的关联和配置环境的易受攻击点进行分析（参见附录一以获得更详细的信息）。IETF RFC 2617描述了在HTTP环境中使用基础验证或消息摘要验证方案时可能遭受的攻击。安全性中可能存在的隐藏风险的影响更应给予特殊的重视。

10.2.2 SAML SOAP 绑定

SOAP 是一个轻量级协议，其目的是在一个分散的、分布式的环境中交换结构化的信息。它使用 XML 技术定义一个可扩展的消息传送框架，提供了 一种可在各种底层协议之上交换的消息结构。该框架的设计独立于任何特定的编程模型和其他的特定语义实现。SOAP 的两个主要设计目标是简单和可扩展。SOAP 试图通过从消息传送框架中省略那些分布式系统中常见的特性来达到上述的目标。这些特性包含但不局限于“可靠性”，“安全性”，“关联”，“路由”和“消息交换模式”（MEP）。

一个 SOAP 消息从根本上说是在 SOAP 节点之间单向传输，它从一个 SOAP 发送端向一个 SOAP 接收端传送，可能通过一个或多个 SOAP 中间媒介进行路由。SOAP 消息与应用相结合，预计能实现更加复杂的交互模式，从请求/应答模式到多方的、来回的“会话式”交换。

SOAP 定义了一个 XML 消息封装，包括消息头段和消息体段，以允许数据和控滞信息的传输。SOAP 还定义了与此封装相关联的操作规则和对于 SOAP 消息传输的 HTTP 绑定。

SAML SOAP 绑定定义了怎样使用 SOAP 来发送和接收 SAML 请求和应答。

与 SAML 类似，SOAP 能在多个底层传输协议之上使用。该绑定具有协议独立性，但在 HTTP 上的 SOAP 使用是必需的（必须实现）。

10.2.2.1 必需的信息

标识：`urn:oasis:names:tc:SAML:2.0:bindings:SOAP`

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：`urn:oasis:names:tc:SAML:1.0:bindings:SOAP-binding`

10.2.2.2 SAML SOAP绑定的协议独立性

以下各节定义 SAML SOAP 绑定独立于底层协议的方面，例如 HTTP 这样的 SOAP 消息在其上传送的协议。该映射仅支持 SOAP 1.1。

10.2.2.2.1 基本操作

SOAP 1.1 消息包含三个元素：封装，头数据和消息体。SAML 请求-应答协议元素必须嵌入在 SOAP 消息体内。

SOAP 1.1 也定义了一个可选的数据编码系统。该系统并不在 SAML SOAP 绑定中使用。这意味着 SAML 消息在从“标准”SAML 方式到基于 SOAP 编码方式，通过 SOAP 传输时，可以无需重新编码。

基于 SOAP 的 SAML 会话系统模型是一个简单的请求-应答模型。

— 一个系统实体作为SAML请求端，传送一条包含在SOAP消息体内的SAML请求元素给另一个作为SAML响应端的系统实体。SAML请求端禁止在每条SOAP消息包含一个以上的SAML请求或在SOAP消息体内包含任何其他的XML元素。

— SAML响应端必须返回一条包含在另一条SOAP消息中的SAML应答元素，或产生一个SOAP故障。SAML响应端禁止在每条SOAP消息包含一个以上的SAML应答或在SOAP消息体内包含任何其他的XML元素。如果SAML响应端因某些原因，不能处理SAML请求，则它必须产生一个SOAP故障。SOAP故障码禁止作为SAML问题域中的错误发送，举例来说，不能找到一个扩展方案或作为在一次授权的查询中主体未被授权访问某个资源的信号。

注（资料性的）— PE19（参见 OASIS PE:2006）建议用下文代替上面的段落：

SAML 响应端应该返回一条 SOAP 消息，该消息包含一个 SAML 应答元素在消息体内，或包含一个 SOAP 故障。SAML 响应端禁止在每条 SOAP 消息包含一个以上的 SAML 应答或在 SOAP 消息体内包含任何其他的 XML 元素。SOAP 故障码禁止作为 SAML 问题域中的错误发送，举例来说，不能找到一个扩展方案或作为在一次授权的查询中主体未被授权访问某个资源的信号。

一旦在一条 SOAP 消息中收到一个 SAML 应答，SAML 请求端不应发送故障码或其他的错误消息给 SAML 响应端。因为消息交互的格式是简单的请求-应答模式，因此增加额外的条目例如错误条件等将不必要的使得协议复杂化。

W3C SOAP 参考了一个 XML Schema 规范的早期草案，该草案包含一个废弃的命名空间。SAML 请求端应该仅参考最新的 XML Schema 命名空间来生成 SOAP 文档。SAML 响应端必须既能够处理用于 SOAP 1.1 （参见 W3C SOAP）的 XML Schema 命名空间，也要能处理最新的 XML Schema 命名空间。

10.2.2.2.2 SOAP头

在 SOAP 上的 SAML 会话中，SAML 请求端可以给 SOAP 消息增加任意的头信息。本绑定不定义任何附加的 SOAP 头。

注 1 — 允许其他头信息的原因是某些SOAP软件和库可能给SOAP消息增加头信息，这些头信息SAML-aware不能处理。此外，某些头信息可能被需要消息路由的底层协议或消息安全机制用到。

SAML 响应端禁止在 SOAP 消息中要求任何头信息，以便其自身正确的处理 SAML 消息，但可以要求附加的标记底层路由或消息安全要求的头信息。

注 2 — 原则上要求额外的头信息会导致SAML标准的分裂，损害互操作性。

10.2.2.3 HTTP上的SOAP的使用

具有SAML SOAP绑定一致性的SAML处理者必须实现基于HTTP的SOAP上的SAML。本节描述使用HTTP上的 SOAP 的一些规定，包括 HTTP 头，缓存和错误报告。

对于 SOAP 的 HTTP 映射在 W3C SOAP 6.0 节中说明。它需要使用 `SOAPAction` 头作为 SOAP HTTP 请求的一部分。SAML 响应端不能依赖于该头信息的值。SAML 请求端设置 `SOAPAction` 头信息值的方式如下：

```
http://www.oasis-open.org/committees/security
```

10.2.2.3.1 HTTP头

在基于 HTTP 的 SOAP 上的 SAML 会话中，SAML 请求端可以给 HTTP 请求增加任意的头信息。本绑定不定义任何附加的 HTTP 头。

注 1 — 允许其他头信息的原因是某些HTTP软件和库可能给HTTP消息增加头信息，这些头信息SAML-aware不能处理。此外，某些头信息可能被需要消息路由的底层协议或消息安全机制用到。

SAML 响应端禁止在 HTTP 请求中要求任何头信息，以便其自身正确的处理 SAML 消息，但可以要求附加的标记底层路由或消息安全要求的头信息。

注 2 — 原则上要求额外的头信息会导致SAML标准的分裂，损害互操作性。

10.2.2.3.2 缓存

HTTP 代理不应缓存 SAML 协议消息。为确保这一点，以下规则应该遵守。

当使用 HTTP1.1 时，请求端应该：

— 包含一个`Cache-Control`头信息字段，设置为“`no-cache, no-store`”。

— 包含一个编译指示头信息字段，设置为“`no-cache`”。

当使用 HTTP1.1 时，响应端应该：

— 包含一个缓冲控制头信息字段，设置为“`no-cache，no-store`，必须使之重新生效，私有”。

— 包含一个编译指示头信息字段，设置为“`no-cache`”。

— 不包含Validator，如`Last-Modified`或ETag头。

10.2.2.3.3 错误报告

拒绝与 SAML 请求端执行消息交换的 SAML 响应端应该返回一个“403 被禁止的”应答。在这种情况下，HTTP 消息体的内容不再有效。

如 W3C SOAP 第 6.2 节所述，在处理 SOAP 请求时出现 SOAP 错误的情况下，SOAP HTTP 服务器必须返回一个“500 内部服务器错误”响应，并且要在应答中包含一个带有 SOAP<SOAP-ENV:fault>元素的 SOAP 消息。这类错误应该在控制权被传给 SAML 处理器之前，或当 SOAP 处理器报告一个内部错误时（例如 SOAP XML 命名空间不正确，SAML Schema 不能被载入，SAML 处理器产生一个异常等等），检测出来作为 SOAP 相关的错误返回。

注（资料性的）— PE19（参见 OASIS Document Errata）建议用下文代替上面段落的第一句：

如 W3C SOAP 第 6.2 节所述，在处理 SOAP 请求时出现 SOAP 错误的情况下，SOAP HTTP 服务器应该返回一个“500 内部服务器错误”响应，并且要在应答中包含一个带有 SOAP<SOAP-ENV:fault>元素的 SOAP 消息。

在 SAML 处理错误的情况下，SOAP HTTP 服务器必须以“200 OK”响应并在 SOAP 消息体内的 SAML 响应中包含一个 SAML-规定的<samlp:Status>元素。<samlp:Status>元素不在 SOAP 消息体内单独出现，只以某种形式出现在 SAML 响应中。

有关 SAML 状态码使用的更多信息，参见本建议书的“SAML 断言和协议”一节。

10.2.2.3.4 元数据考虑因素

对于 SOAP 映射的支持反映为指示一个 URL 端点，在那里待发送的请求包含在特定协议或配置的 SOAP 消息中，也可选择给出 WSDL 端口/端点的定义。

10.2.2.3.5 使用HTTP上的SOAP交换SAML消息的例子

以下是从 SAML 属性授权中心处请求一个包含属性断言断言的查询的例子。

```
POST /SamlService HTTP/1.1
Host: www.example.com
Content-Type: text/xml
Content-Length: nnn
SOAPAction: http://www.oasis-open.org/committees/security
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV=”http://schemas.xmlsoap.org/soap/envelope/”>
    <SOAP-ENV:Body>
        <samlp:AttributeQuery xmlns:samlp:=”…”
xmlns:saml=”…” xmlns:ds=”…” ID=”_6c3a4f8b9c2d” Version=”2.0”
IssueInstant=”2004-03-27T08:41:00Z”
            <ds:Signature> … </ds:Signature>
            <saml:Subject>
            …
            </saml:Subject>
        </samlp:AttributeQuery>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
Following is an example of the corresponding response, which supplies an
assertion containing the attribute statement as requested.
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: nnnn
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV=”http://schemas.xmlsoap.org/soap/envelope/”>
    <SOAP-ENV:Body>
        <samlp:Response xmlns:samlp=”…” xmlns:saml=”…” xmlns:ds=”…”
ID=”_6c3a4f8b9c2d” Version=”2.0” IssueInstant=”2004-03-27T08:42:00Z”>
            <saml:Issuer>https://www.example.com/SAML</saml:Issuer>
            <ds:Signature> … </ds:Signature>
            <Status>
              <StatusCode Value=”…”/>
            </Status>
```

```
            <saml:Assertion>
                <saml:Subject>
                …
                </saml:Subject>
                <saml:AttributeStatement>
            …
                </saml:AttributeStatement>
             </saml:Assertion>
        </samlp:Response>
    </SOAP-Env:Body>
</SOAP-ENV:Envelope>
```

10.2.3 反向SOAP（PAOS）绑定

本绑定是 SOAP 规范的反向 HTTP 绑定（参见 PAOS:2003）。除了本建议书规定的规则之外，实现还必须满足 PAOS 规定的一般性操作规则。如果规则有冲突的话，以 Liberty Alliance POAS:2003 为准。

10.2.3.1 必需的信息

标识：`urn:oasis:names:tc:SAML:2.0:bindings:PAOS`

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.3.2 概述

反向 SOAP 绑定是一种机制，通过它 HTTP 请求端能够对 SAML 请求端通告其作为 SOAP 响应端或 SOAP 媒介的能力。HTTP 请求端能支持一种模式，在该模式下 SAML 请求端可以给它发送一个在 HTTP 应答中嵌入在 SOAP 封装里的 SAML 请求，HTTP 请求端在随后的 HTTP 请求中以一个嵌入在 SOAP 封装中的 SAML 响应作为应答。这种消息交换模式支持在 ECP SSO 配置中定义的使用情况，在该配置中 HTTP 请求端在认证交换中作为中间媒介。

10.2.3.3 消息交换

PAOS 映射包含两种消息交换模式组合：

1） HTTP请求端发送HTTP请求给SAML请求端。SAML请求端以一个HTTP响应作为应答，该响应包含嵌入了SAML请求消息的SOAP封装。

2） 随后，HTTP请求端发送一个HTTP请求给起始SAML请求端，该请求包含一个嵌入了SAML响应消息的SOAP封装。SAML请求端以一个HTTP响应作为应答，或者在步骤1中响应起始服务器请求。

ECP 配置使用 PAOS 映射让服务提供商在提供服务前对客户方进行认证。其步骤如下，图 10-1 所示。

1） 客户方使用HTTP请求来请求某项服务。

2） 服务提供商响应以SAML认证请求。该请求使用SOAP请求，以HTTP应答作为载体发送。

3） 客户方返回一个带有SAML认证响应的SOAP应答。该应答使用一个新的HTTP请求发送。

4） 假定服务提供商认证并授权成功，则服务提供商可以给起始的服务请求以HTTP响应做出应答。

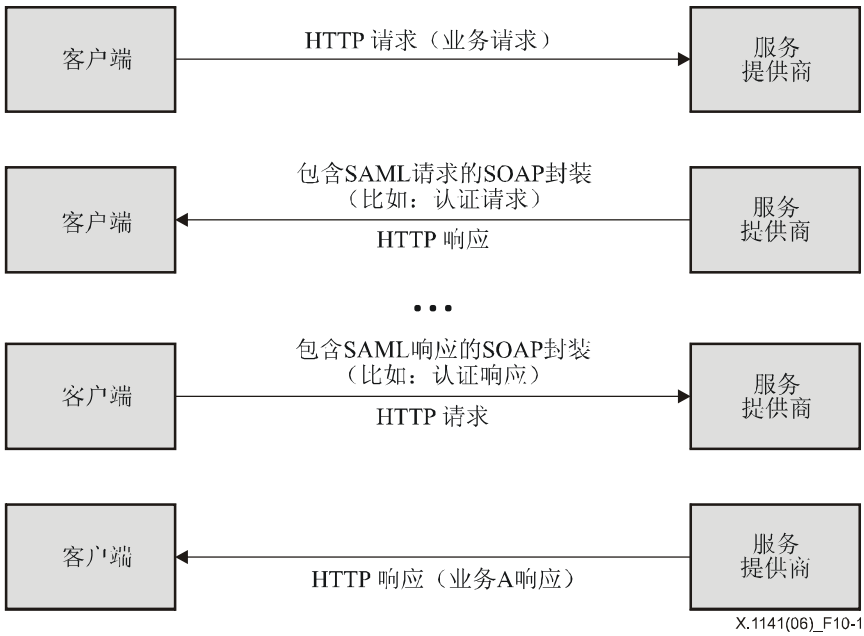

图 10-1/X.1141－PAOS映射消息交换

HTTP 请求端在其 HTTP 请求中使用 PAOS:2003 规范定义的 HTTP 头信息来通告其处理反向 SOAP 绑定的能力。规定如下：

- HTTP `Accept` Header字段必须指出其接受“`application/vnd.paos+xml`”内容形式的能力。
- HTTP `PAOS` Header字段必须出现并且至少规定PAOS版本为“`urn:liberty:paos:2003-08`”。

注 1（资料性的）— PE21（参见OASIS PE:2006）建议从上文中删除“至少”。

其他的 PAOS 头信息如服务取值等可能在使用 PAOS 映射的配置中规定。HTTP 请求端可以对 HTTP 请求添加任意的头信息。

注 2 — 本绑定未定义RelayState机制。因此使用本绑定的特定配置如果需要，则必须定义这样的机制。为达到此目的，建议使用SOAP头。

以下各节对于消息交换的两个步骤提供了更多细节说明。

10.2.3.3.1 HTTP请求，SOAP响应中的SAML请求

为了响应任意的 HTTP 请求，HTTP 响应端可以通过返回一个在 HTTP 响应中的 SOAP1.1 封装来使用该映射返回一个 SAML 请求消息，在 SOAP 消息体中仅包含单一的 SAML 请求消息，不包括其他内容。SOAP 封装可以包含 PAOS、SAML 配置或其他建议定义的任意 SOAP 头信息。

由于 SAML 请求消息是传递到 HTTP 请求端，而实际的预期接收端可能是另一个系统实体，因此 HTTP 请求端如特定配置定义的那样，可能充当的是一个 中间媒介。

10.2.3.3.2 SOAP请求中的SAML响应，HTTP响应

当 HTTP 请求端使用 PAOS 绑定传递 SAML 响应消息给预期的接收端时，它将该消息作为 HTTP 请求中嵌入在 SOAP 封装里的 SOAP 消息体的唯一元素。HTTP 请求端可以是也可以不是 SAML 响应的发起方。SOAP 封装可以包含 PAOS、SAML 配置或其他建议书定义的任意 SOAP 头信息。SAML 交换被看做是完备的，HTTP 响应本绑定未特别规定。

协议子集可以定义本绑定所覆盖的交换中非 SOAP 响应部分的 HTTP 内容的其他限定。

10.2.3.4 缓存

HTTP 代理不应缓存 SAML 协议消息。为确保这一点，以下规则应该遵守。

当使用 HTTP1.1 时，发送 SAML 协议消息的请求端应该：

- 包含一个缓存控制头信息字段，设置为“no-cache, no-store”。
- 包含一个编译指示头信息字段，设置为“no-cache”。

当使用 HTTP1.1 时，返回 SAML 协议消息的响应端应该：

- 包含一个缓冲控制头信息字段，设置为“no-cache, no-store，必须使之重新生效，私有”。
- 包含一个编译指示头信息字段，设置为“no-cache”。
- 不包含Validator，如Last-Modified或ETag头。

10.2.3.5 安全性考虑

在 PAOS 绑定中的 HTTP 请求端可以作为 SOAP 中间媒介，此时，对于起始认证的传输层安全性，完整性和可信度的传输层安全性不能满足端到端的安全性要求。这种情况下建议使用 SOAP 消息层的安全机制。

注（资料性的）— PE31（参见OASIS PE:2006）建议将recommended改为RECOMMENDED。

10.2.3.5.1 错误报告

标准 HTTP 和 SOAP 错误约定必须被遵守。SAML 处理过程中出现的错误禁止在 HTTP 或 SOAP 层被标记，并且必须使用带有错误<samlp:Status>元素的 SAML 响应消息来处理。

10.2.3.5.2 元数据考虑因素

对于 PAOS 绑定的支持反映为指明一个 URL 端点，在那里针对特定协议或配置的嵌入在 SOAP 封装中的 HTTP 请求和/或 SAML 协议消息等待发送。也可提供单独的端点或不同的请求和响应端点。

10.2.4 HTTP重定向绑定

HTTP 重定向绑定定义了一种机制，通过它 SAML 协议消息能以 URL 参数传输。可容许的 URL 长度理论上没有限制，但实际上不可预知的受到限制。因此，需要专门的编码来承载 URL 上的 XML 消息，更大或更复杂的消息内容需要使用 HTTP POST 或 Artifact 映射来发送。

本绑定可以由 HTTP POST 绑定（参见第 10.2.5 节）和 HTTP Artifact 绑定（参见第 10.2.6 节）组成，使用两个不同的映射在一个单独的协议交换中传输请求和响应消息。

本绑定使用消息编码。一旦本绑定的定义中包含一个特定消息编码的定义，则其余的可以被定义和使用。

10.2.4.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.4.2 概述

HTTP 重定向绑定的目的是为处理 SAML 请求端和响应端需要使用 HTTP 用户代理（如 IETF RFC 2616 中定义）作为中间媒介进行通信的情况。这是必要的，例如通信的各方没有共享一条直接的通信路径。如果响应端需要与用户代理进行交互以完成请求，例如当用户代理必须对其进行认证时，也需要使用该映射。

某些 HTTP 用户代理能够在协议交换中起到更加积极的作用，并且可以支持其他的使用 HTTP 的映射，例如 SOAP 和反向 SOAP 绑定。本绑定假定用户代理只具备普通网络浏览器的功能。

10.2.4.3 RelayState

本绑定传输的 SAML 协议消息中可以包含 RelayState 数据。数据值在长度上不能超过 80 个字节，并且应该独立于其他在消息传输期间可能存在或可能不存在的保护，由创建消息的实体完整保护。鉴于空间的限制，签字是不可行的，但由于该数据值暴露于第三方的干扰中，因此实体应该采用校验和，伪随机值或其他类似的手段来确保该数据值不被篡改。

注（资料性的）— PE1（参见OASIS PE:2006）断言，上段的最后一句应以如下方式理解：

本绑定传输的 SAML 协议消息中可以包含 RelayState 数据。数据值在长度上不能超过 80 个字节，并且应该由创建消息的实体通过数字签字（参见第 10 节）或某些独立的方式完整保护。

如果一个 SAML 请求消息带有 RelayState 数据，则 SAML 响应端必须采用一个同样支持 RelayState 机制的映射来返回其 SAML 协议响应，并且必须将其在请求中接收到的准确的 RelayState 数据置于响应中相应的 RelayState 参数中。

如果一个 SAML 请求消息中没有包含该数据值，或 SAML 响应消息在没有对应请求的情况下生成，那么 SAML 响应端可以包含 RelayState 数据，交由接收端基于协议子集或各方之间的预先约定来进行解释。

10.2.4.4 消息编码

本绑定对消息采用 URL 编码技术进行编码，使用 HTTP GET 方法进行传输。有许多可能的方法将 XML 编码成 URL，采用何种方法取决于效果的限制。本建议书定义一种这样的方法，而并不排除其他的方法。映射端点应该使用元数据适时指出其支持的编码方式。专用编码在定义时必须使用一个 URI 唯一标识。并不要求所有的 SAML 消息都能用一组特定的规则进行编码，但是规则必须清楚地指出哪些消息或内容能够或者不能够被这样编码。

一个 URL 编码必须将消息完全放在 URL 查询字符串中，并且必须为消息接收端的端点保留 URL 的其余部分。

称作 `SAMLEncoding` 的查询字符串参数留做标识查询使用的编码机制。如果该参数被省略，那么则该值被假定为 `urn:oasis:names:tc:SAML:2.0:bindings:URL-Encoding:DEFLATE`。

所有支持本绑定的端点都必须支持下文所述的 DEFLATE 编码。

i） DEFLATE 编码

标识：`urn:oasis:names:tc:SAML:2.0:bindings:URL-Encoding:DEFLATE`

SAML 协议消息可以通过 DEFLATE （IETF RFC 1951 ）压缩方法编码成一个 URL。在该编码中，对起始 SAML 协议消息的 XML 序列应用以下的过程：

1） 必须删除SAML协议消息上的任何签字，包括`<ds:Signature>` XML元素自身。如果消息内容中包括另一个签字，例如一个签字的SAML断言，则不删除这个嵌入的签字。然而，这样的一条消息在编码后的长度已经排除了使用该编码机制的可能性。因此包含签字内容的SAML协议消息不应采用该机制进行编码。

2） 对于起始SAML协议消息所剩下的全部XML内容采用IETF RFC 1951中规定的DEFLATE压缩机制。

3） 压缩后的数据随后按照IETF RFC 2045中规定的规则进行基64编码。换行或其他的空白必须从结果中删除。

4） 基64编码数据再进行URL编码，并且作为一个查询字符串参数加到URL中，该参数必须命名为`SAMLRequest`（如果消息是一个SAML请求）或`SAMLResponse`（如果消息是一个SAML应答）。

5） 如果SAML协议消息中带有RelayState数据，则该数据必须进行URL编码并且放在另一个查询字符串参数中，称为`RelayState`。

6） 如果起始SAML协议消息使用了XML数字签字，则必须使用下述规则附加一个包含上述编码数据的新签字。

XML 数字签字由于空间的原因，并不直接按照上述规则进行 URL-编码。如果 SAML 协议消息带有 XML 签字，则其 URL-编码形式必须按下面的描述签字：

1） 必须包含一个附加的查询字符串参数作为签字算法标识符，命名为`SigAlg`。该参数的取值必须是一个URI，它标识用来对URL-编码的SAML协议消息进行签字的算法，遵从XML签字或其他建议管理的算法规定。

2） 为构建该签字，一个由`RelayState`（如果有的话），`SigAlg`和`SAMLRequest`（或`SAMLResponse`）查询字符串参数（每个都进行了URL-编码）串连组成的字符串按以下方式合并为一（次序如下）：

a） `SAMLRequest=value&RelayState=value&SigAlg=value`
`SAMLResponse=value&RelayState=value&SigAlg=value`

b） 结果字符串在字节形式上是八位组字符串，以适应数字签字算法。起始查询字符串的任何其他内容都不包含在其中，也不进行签字。

c） 签字值必须在删除空白后使用基64编码（参见IETF RFC 2045）进行编码，然后作为一个查询字符串参数被包含，参数名为`Signature`。在基64编码签字值中的一些字符本身在添加之前可能需要URL-编码。

d） 本编码机制必须支持以下签字算法（参见W3C 签字）及其URI表示：

- DSAwithSHA1 – http://www.w3.org/2000/09/xmldsig#dsa-sha1
- RSAwithSHA1 – http://www.w3.org/2000/09/xmldsig#rsa-sha1

注 — NIST（国际标准和技术协会）现在鼓励使用SHA-256（256比特编码密钥的安全哈希算法）代替SHA-1。

当验证签字时，本绑定并不规定待验证的结果 URL 上的查询字符串参数顺序。参数可以任意顺序出现。在验证一个签字之前，信任方必须保证待验证的参数值按照上述签字规则的要求排列。

URL-编码并不是唯一规范，也就是说，对于给定的值，有多种合法的编码方法。因此信任方必须使用其在查询字符串中接收到的起始 URL-编码值实施验证步骤。在参数被软件处理后对其进行重编码是不够的，因为结果编码可能与签字者的编码不匹配。

如果没有 `RelayState` 数据值，签字计算中应该完全省略该参数（并且作为一个空参数名，不包含在内）。

10.2.4.5 消息交换

通过本绑定进行的 SAML 会话所使用的系统模型是一个请求-响应模型，但这些消息是在 HTTP 响应中发送给用户代理，在 HTTP 请求中传送给消息接收方的。这些交换发生之前、之间以及之后的 HTTP 交互是隐含的。SAML 请求端和 SAML 响应端都假定是 HTTP 响应端。下面的顺序图表（图 10-2）解释了消息交换的过程。

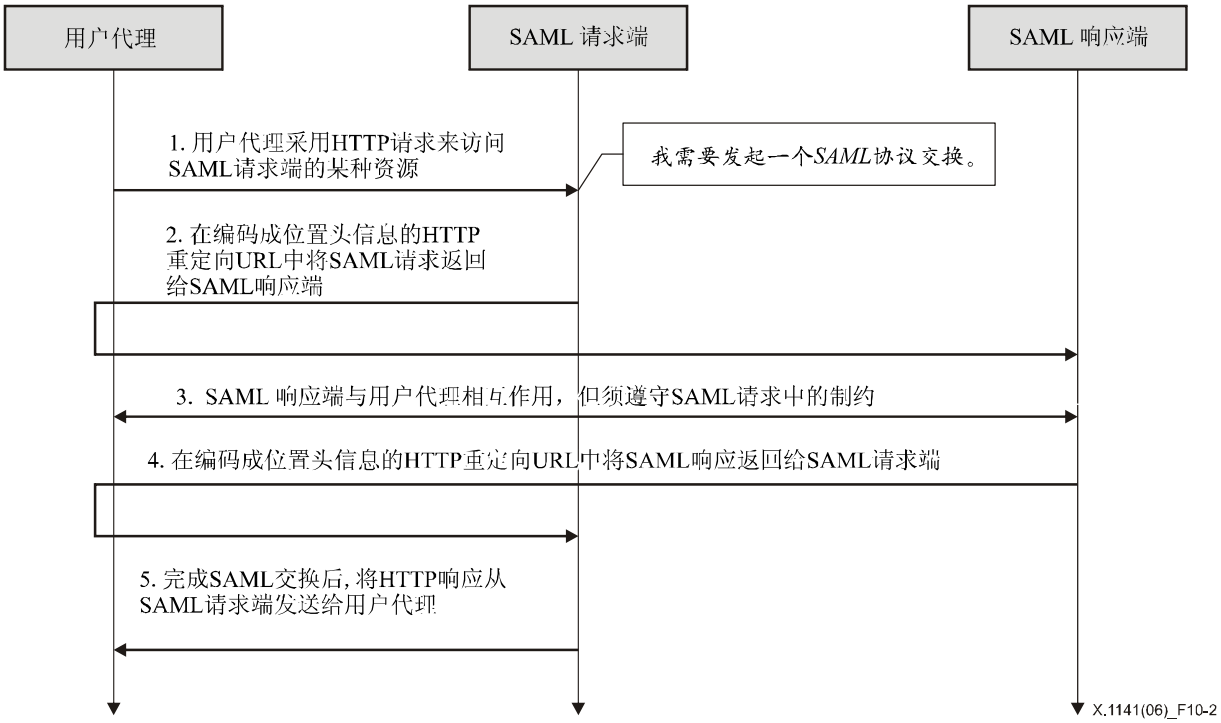

图 10-2/X.1141－HTTP重定向消息交换

1） 首先，用户代理对系统实体做出任意的HTTP请求。在处理该请求的过程中，系统实体决定启动一次SAML协议交换。

2） 系统实体作为SAML请求端对步骤1中用户代理提出的HTTP请求返回一个SAML请求作为应答。该SAML请求被编码成HTTP响应的Location头返回，其HTTP状态必须设为303或302。SAML请求端可以在HTTP响应中包含其他的内容和表达以便用户代理的消息传输，如IETF RFC 2616中定义的那样。用户代理通过发布一个HTTP GET请求给SAML响应端来传输SAML请求。

3） 一般说来，SAML响应端可以立即返回一条SAML响应给SAML请求作为应答，也可以返回任意的内容以便与完成请求所必需的用户代理进行后续交互。特定的协议和协议子集可能包含请求端容许此类交互的主动性程度的指示机制（举例来说，`<samlp:AuthnRequest>`中的`IsPassive`属性）。

4） 最终响应端应该返回一条SAML响应给用户代理，以返回给SAML请求端。SAML响应与步骤2中所述的SAML请求的方式一样返回。

5） 一收到SAML响应，SAML请求端即返回任意的一条HTTP响应给用户代理。

10.2.4.5.1 HTTP及缓存考虑

HTTP 代理和用户代理中介不应缓存 SAML 协议消息。为确保这一点，以下规则应该遵守：

当使用 HTTP1.1 返回 SAML 协议消息时，HTTP 响应端应该：

- 包含一个缓存控制头信息字段，设置为“`no-cache, no-store`”。
- 包含一个编译指示头信息字段，设置为“`no-cache`”。

对于 HTTP 头的使用没有其他限制。

10.2.4.5.2 安全性考虑

用户代理中介的参与意味着请求端和响应端不能依赖传输层保证端到端的认证、完整性和可信度。如果编码方法规定了一种签字手段，那么 URL-编码消息可以通过签字来提供起始验证和完整性。

如果消息是签字的，在协议消息根 SAML 元素中的 Destination XML 属性必须包含发送方命令用户代理传送消息的 URL。而接收方必须验证该值与该消息接收地址是否匹配。

如果请求或响应的内容不应暴露给用户代理中介，则不应使用本绑定。否则，SAML 请求和 SAML 响应二者的信任度是任意的，且依赖于使用环境。如果信任度是必需的，则应使用 TLS1.0 来保护在用户代理和 SAML 请求端和响应端之间传递的消息。

URL-编码消息可以在各种 HTTP 日志和 HTTP “Referrer”头中出现。

认证、消息完整性和可信度机制的每种组合在配置之前，应该先针对特定协议交换的背景和配置环境的易受攻击点进行分析（参见附录一）。

总之，本绑定依赖于消息级的认证以及通过签字方式的完整性保护，并且不支持来自用户代理中介的消息可信度。

10.2.4.6 错误报告

拒绝与 SAML 请求端执行消息交换的 SAML 响应端应该返回一个带有 urn:oasis:names:tc:SAML:2.0: status:RequestDenied 的第二级<samlp:StatusCode>值的 SAML 响应消息。

消息交换期间的 HTTP 交互禁止使用 HTTP 错误状态码来指示 SAML 处理中的失败，因为用户代理对于 SAML 协议交换来说不是完整的一方。也见第 9 节。

10.2.4.7 元数据方面的考虑

对于 HTTP 重定向绑定的支持反映为指明 URL 端点，在那里应该发送针对特定协议或配置的请求和响应。也可提供单独的端点或不同的请求和响应端点。

注（资料性的）— PE2（参见OASIS PE:2006）断言用以下内容代替上段：

对于使用HTTP Artifact绑定接收消息的支持反映为指明URL端点，在那里应该发送针对特定协议或配置的请求和响应。也可提供单独的端点或不同的请求和响应端点。对于使用本绑定发送消息的支持应该伴随有一个或多个编有索引的<md:ArtifactResolutionService>端点以处理<samlp:ArtifactResolve>消息。

10.2.4.8 使用HTTP重定向的SAML消息交换举例

本例中，一个<LogoutRequest>和 <LogoutResponse>消息对使用 HTTP 重定向绑定进行交换。

首先是将要交换的实际的 SAML 协议消息：

```
<samlp:LogoutRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="d2b7c388cec36fa7c39c28fd298644a8"
IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://IdentityProvider.com/SAML</Issuer>
    <NameID
Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">005a06e0-ad82
-110d-a556-004005b13a2b</NameID>
    <samlp:SessionIndex>1</samlp:SessionIndex>
</samlp:LogoutRequest>

<samlp:LogoutResponse xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="b0730d21b628110d8b7e004005b13a2b"
InResponseTo="d2b7c388cec36fa7c39c28fd298644a8"
    IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://ServiceProvider.com/SAML</Issuer>
    <samlp:Status>
        <samlp:StatusCode
Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
```

```
    </samlp:Status>
</samlp:LogoutResponse>
```

本绑定不对步骤 1 中从上面用户代理处发出的初始 HTTP 请求进行定义。为发起注销协议交换，SAML 请求端返回以下 HTTP 响应，包含一个签字的 SAML 请求消息。SAMLRequest 参数值实际上来自上面的请求消息。签字部分仅列举出来而不作为实际计算的结果。填入下面的 HTTP Location 头中的那一行是文档人造的，实际的头中不填入数据值。

```
HTTP/1.1 302 Object Moved
Date: 21 Jan 2004 07:00:49 GMT
Location:
https://ServiceProvider.com/SAML/SLO/Browser?SAMLRequest=fVFdS8MwFH0f7D%2BU
vGdNsq62oSsIQyhMESc%2B%2BJYlmRbWpObeyvz3puv2IMjyFM7HPedyK1DdsZdb%2F%2BEHfLF
fgwVMTt3RgTwzazIEJ72CFqRTnQWJWu7uH7dSLJjsg0ev%2FZFMlttiBWADtt6R%2BSyJr9msiR
H7O70sCm31Mj%2Bo%2BC%2B1KA5GlEWeZaogSQMw2MYBKodrIhjLKONU8FdeSsZkVr6T5M0GiHM
jvWCknqZXZ2OoPxF7kGnaGOuwxZ%2Fn4L9bY8NC%2By4du1XpRXnxPcXizSZ58KFTeHujEWkNPZ
ylsh9bAMYYUjO2Uiy3jCpTCMo5M1StVjmN9SO150sl9lU6RV2Dp0vsLIy7NM7YU82r9B90PrvCf
85W%2FwL8zSVQzAEAAA%3D%3D&RelayState=0043bfc1bc45110dae17004005b13a2b&SigAl
g=http%3A%2F%2Fwww.w3.org%2F200%2F09%2Fxmldsig%23rsa-sha1&Signature=NOTAREA
LSIGNATUREBUTTHEREALONEWOULDGOHERE
Content-Type: text/html; charset=iso-8859-1
```

在任何可能的不特定的交互发生之后，SAML 响应端返回如下的 HTTP 响应，包含签字的 SAML 响应消息。再次的，SAMLResponse 参数值实际上来自上面的响应消息。签字部分仅列举出来而不作为实际计算的结果。

```
HTTP/1.1 302 Object Moved
Date: 21 Jan 2004 07:00:49 GMT
Location:
https://IdentityProvider.com/SAML/SLO/Response?SAMLResponse=fVFNa4QwEL0X%2B
h8k912TaDUGFUp7EbZQ6rKH3mKcbQVNJBOX%2FvxaXQ9tYec0vHlv3nzkqIZ%2BlAf7YSf%2FBj
hagxB8Db1BuZQKMjkjrcIOpVEDoPRa1o8vB8n3VI7OeqttT1bJbbJCBOc7a8j9XTBH9VyQhqYRb
TlrEi4Yo61oUqA0pvShYZHiDQkqs411tAVpeZPqSAgNOkrOas4zzcW55ZlI4liJrTXiBJVBr4wv
CJ877ijbcXZkmaRUxtk7CU7gcB5mLu8pKVddvghd%2Ben9iDIMa3CXTsOrs5euBbfXdgh%2F9sn
DK%2FEqW69Ye%2BUnvGL%2F8CfbQnBS%2FQS3z4QLW9aT1oBIws0j%2FGOyAb9%2FV34Dw5k779
IBAAA%3D&RelayState=0043bfc1bc45110dae17004005b13a2b&SigAlg=http%3A%2F%2Fww
w.w3.org%2F200%2F09%2Fxmldsig%23rsa-sha1&Signature=NOTAREALSIGNATUREBUTTHER
EALONEWOULDGOHERE
Content-Type: text/html; charset=iso-8859-1
```

10.2.5 HTTP POST绑定

HTTP POST 绑定定义了一种机制，通过它 SAML 协议消息能在 HTML 表单控件命令的基 64 编码内容中传输。

本绑定可以由 HTTP 重定向绑定（参见第 10.2.4 节）和 HTTP Artifact 绑定（参见第 10.2.6 节）组成，使用两个不同的映射在一个单独的协议交换中传输请求和响应。

10.2.5.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.5.2 概述

HTTP POST 绑定的目的是为处理 SAML 请求端和响应端需要使用 HTTP 用户代理（如 IETF RFC 2616 中定义）作为中间媒介进行通信的情况。这是必要的，例如通信的各方没有共享一条直接的通信路径。如果响应端需要与用户代理进行交互以完成请求，例如当用户代理必须对其进行认证时，也需要使用该映射。

某些 HTTP 用户代理能够在协议交换中起到更加积极的作用，并且可以支持其他的使用 HTTP 的映射，例如 SOAP 和反向 SOAP 绑定。本绑定假定用户代理只具备普通网络浏览器的功能。

10.2.5.3 RelayState

本绑定传输的 SAML 协议消息中可以包含 RelayState 数据。值在长度上不能超过 80 个字节，并且应该独立于其他在消息传输期间可能存在或可能不存在的保护，由创建消息的实体完整保护。鉴于空间的限制，签字是不可行的，但由于该数据值暴露于第三方的干扰中，因此实体应该采用校验和，伪随机值或其他类似的手段来确保该数据值不被篡改。

如果一个 SAML 请求消息带有 RelayState 数据，则 SAML 响应端必须采用一个同样支持 RelayState 机制的映射来返回其 SAML 协议响应，并且必须将其在请求中接收到的准确的 RelayState 数据置于响应中相应的 RelayState 参数中。

如果一个 SAML 请求消息中没有包含该数据值，或 SAML 响应消息在没有对应请求的情况下生成，那么 SAML 响应端可以包含 RelayState 数据，交由接收端基于协议子集或各方之间的预先约定来进行解释。

注（资料性的）— PE31（参见OASIS PE:2006）建议以如下内容阐明上段：

如果一个 SAML 请求消息中没有包含 RelayState 参数，或 SAML 响应消息在没有对应请求的情况下生成，那么 SAML 响应端可以包含 RelayState 数据，交由接收端基于协议子集或各方之间的预先约定来进行解释。

10.2.5.4 消息编码

本绑定使用的消息通过将 XML 编码成为 HTML 表单控件命令的方式进行编码，使用 HTTP POST 方法进行传输。一条 SAML 协议消息通过对消息的 XML 表示采用基 64 编码规则编码，并以 W3C HTML 第 17 节中定义的表单将结果置入一个隐含的表单控件命令中，来进行格式化编码。HTML 文件必须支持 W3C XHTML 的常规实例。

如果消息是 SAML 请求，则表单控件命令必须命名为 `SAMLRequest`。如果消息是一个 SAML 响应，则表单控件命令必须命名为 `SAMLResponse`。可以包含其他的表单控件命令或表征，但它们不能是接收方处理消息所必需的。

如果 SAML 协议消息中伴随有“RelayState”值，则它必须被放在另一个隐含表单控件命令中，称为 `RelayState`，与 SAML 消息的表单相同。

表单的行为属性必须是接收方的 HTTP 端点，对使用本绑定的协议或协议子集来说，SAML 消息就传送到这个端点。方法属性必须是“POST”。

可以使用用户代理所支持的任何技术来提交表单，而任何支持这一点所需的表单内容都可以被包含，例如服从控制和客户端脚本命令。然而，接收端必须能够不考虑这种表单提交的机制来处理消息。

任何包含的表单控件值必须被转换以便安全的包含在 XHTML 文件中。这包括转换诸如引号之类的字符成为 HTML 实体类的特征。

10.2.5.5 消息交换

通过本绑定进行的 SAML 会话所使用的系统模型是一个请求-响应模型，但这些消息是在 HTTP 响应中发送给用户代理，在 HTTP 请求中传送给消息接收端的。这些交换发生之前、之间以及之后的 HTTP 交互是隐含的。SAML 请求端和 SAML 响应端都假定是 HTTP 响应端。图 10-3 表明了交换的消息。

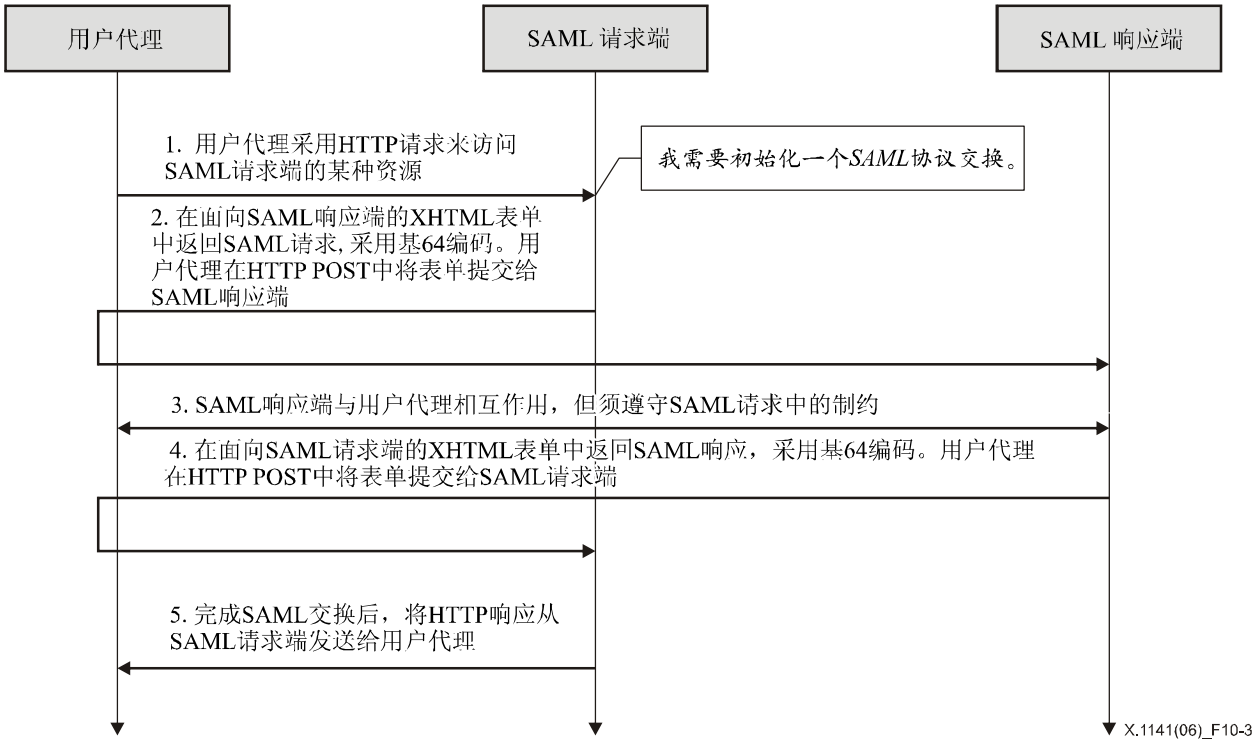

图 10-3/X.1141－HTTP POST 消息交换

1） 首先，用户代理对系统实体做出任意的HTTP请求。在处理该请求的过程中，系统实体决定启动一次SAML协议交换。

2） 系统实体作为SAML请求端对用户代理提出的HTTP请求返回一个SAML请求作为应答。该请求在一个XHTML文件中返回，文件中包含第10.2.5.4节定义的表单和内容。用户代理通过发布一个HTTP POST请求给SAML响应端来传输SAML请求。

3） 一般说来，SAML响应端可以立即返回一条SAML响应给SAML请求作为应答，也可以返回任意的内容以便与完成请求所必需的用户代理进行后续交互。特定的协议和协议子集可能包含请求端容许此类交互的主动性程度的指示机制（举例来说，`<samlp:AuthnRequest>`中的`IsPassive`属性）。

4） 最终响应端应该返回一条SAML响应给用户代理，以返回给SAML请求端。SAML响应与步骤2中所述的SAML请求的方式一样返回。

5） 一收到SAML响应，SAML请求端即返回任意的一条HTTP响应给用户代理。

10.2.5.5.1 HTTP及缓存考虑

HTTP 代理和用户代理中介不应缓存 SAML 协议消息。为确保这一点，以下规则应该遵守：

当使用 HTTP1.1 返回 SAML 协议消息时，HTTP 响应端应该：

- 包含一个缓存控制头信息字段，设置为“`no-cache, no-store`”。
- 包含一个编译指示头信息字段，设置为“`no-cache`”。

对于 HTTP 头的使用没有其他限制。

10.2.5.5.2 安全性考虑

用户代理中介的参与意味着请求端和响应端不能依赖传输层保证端到端的验证、完整性和可信度，而必须代以接收消息的认证。SAML 为这类情况下协议消息上签字的认证和完整性作了准备。格式化编码的消息可以在基 64 编码应用之前进行签字。

如果消息是签字的，在协议消息根 SAML 元素中的 Destination XML 属性必须包含发送方命令用户代理传送消息的 URL。而接收方必须验证该值与该消息接收地址是否匹配。

如果请求或响应的内容不应暴露给用户代理中介，则不应使用本绑定。否则，SAML 请求和 SAML 响应二者的信任度是任意的，且依赖于使用环境。如果信任度是必需的，则应使用 TLS1.0 来保护在用户代理和 SAML 请求端以及响应端之间传递的消息。

总之，本绑定依赖于消息级的认证以及通过签字方式的完整性保护，并且不支持来自用户代理中介的消息可信度。

没有定义一种机制来保护 SAML 协议消息和如果存在的“RelayState”值之间关系的完整性。也就是说，攻击者可能通过转换与每个 SAML 协议消息相关联的“RelayState”值来重组一对有效的 HTTP 响应。单个的“RelayState”和 SAML 消息可以进行完整性保护，而其组合则不行。结果，“RelayState”信息的产生者和消费者必须注意不要不加以额外的预防措施（例如基于 SAML 消息中的信息）就将敏感的状态信息与“RelayState”值相结合。

10.2.5.6 错误报告

拒绝与 SAML 请求端执行消息交换的 SAML 响应端应该返回一个带有 urn:oasis:names:tc:SAML:2.0:status:RequestDenied 的第二级<samlp:StatusCode>值的响应消息。

消息交换期间的 HTTP 交互禁止使用 HTTP 错误状态码来指示 SAML 处理中的失败，因为用户代理对于 SAML 协议交换来说不是完整的一方。

更多关于 SAML 状态码的信息，参见第 8.2 节。

10.2.5.7 元数据方面的考虑

对于 HTTP POST 绑定的支持应该反映为指明 URL 端点，在那里应该发送针对特定协议或配置的请求和响应。也可提供单独的端点或不同的请求和响应端点。

10.2.5.8 使用HTTP POST的SAML消息交换举例

本例中，一个<LogoutRequest>和 <LogoutResponse>消息对使用 HTTP POST 绑定进行交换。

首先是将要交换的实际的 SAML 协议消息：

```
<samlp:LogoutRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="d2b7c388cec36fa7c39c28fd298644a8"
IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://IdentityProvider.com/SAML</Issuer>
    <NameID
Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">005a06e0-ad82
-110d-a556-004005b13a2b</NameID>
    <samlp:SessionIndex>1</samlp:SessionIndex>
</samlp:LogoutRequest>

<samlp:LogoutResponse xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="b0730d21b628110d8b7e004005b13a2b"
InResponseTo="d2b7c388cec36fa7c39c28fd298644a8"
    IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://ServiceProvider.com/SAML</Issuer>
    <samlp:Status>
        <samlp:StatusCode
Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
    </samlp:Status>
</samlp:LogoutResponse>
```

本绑定不对步骤 1 中从用户代理处发出的初始 HTTP 请求进行定义。为发起注销协议交换，SAML 请求端返回以下 HTTP 响应，包含一个 SAML 请求消息。SAMLRequest 参数值实际上来自上面的请求消息。

```
HTTP/1.1 200 OK
Date: 21 Jan 2004 07:00:49 GMT
Content-Type: text/html; charset=iso-8859-1

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<body onload="document.forms[0].submit()">

<noscript>
<p>
<strong>Note:</strong> Since your browser does not support JavaScript, you must
press the Continue button once to proceed.
</p>
</noscript>

<form action="https://ServiceProvider.com/SAML/SLO/Browser" method="post">
<div>
<input type="hidden" name="RelayState"
value="0043bfc1bc45110dae17004005b13a2b"/>
<input type="hidden" name="SAMLRequest"
value="PHNhbWxwOkxvZ291dFJlcXVlc3QgeG1sbnM6c2FtbHA9InVybjpvYXNpczpuYW1l
czp0YzpTQU1MOjIuMDpwcm90b2NvbCIgeG1sbnM9InVybjpvYXNpczpuYW1lczp0
YzpTQU1MOjIuMDphc3NlcnRpb24iDQogICAgSUQ9ImQyYjdjMzg4Y2VjMzZmYTdj
MzljMjhmZDI5ODY0NGE4IiBJc3N1ZUluc3RhbnQ9IjIwMDQtMDEtMjFUMTk6MDA6
NDlaIiBWZXJzaW9uPSIyLjAiPg0KICAgIDxJc3N1ZXI+aHR0cHM6Ly9JZGVudGl0
eVByb3ZpZGVyLmNvbS9TQU1MPC9Jc3N1ZXI+DQogICAgPE5hbWVJRCBGb3JtYXQ9
InVybjpvYXNpczpuYW1lczp0YzpTQU1MOjIuMDpuYW1laWQtZm9ybWF0OnBlcnNp
c3RlbnQiPjAwNWEwNmUwLWFkODItMTEwZC1hNTU2LTAwNDAwNWIxM2EyYjwvTmFt
ZUlEPg0KICAgIDxzYW1scDpTZXNzaW9uSW5kZXg+MTwvc2FtbHA6U2Vzc2lvbklu
ZGV4Pg0KPC9zYW1scDpMb2dvdXRSZXF1ZXN0Pg=="/>
</div>
<noscript>
<div>
<input type="submit" value="Continue"/>
</div>
</noscript>
</form>
</body>
</html>
```

在任何可能的不特定的交互发生之后，SAML 响应端返回如下的 HTTP 响应，包含 SAML 响应消息。再次的，SAMLResponse 参数值实际上来自上面的响应消息。

```
HTTP/1.1 200 OK
Date: 21 Jan 2004 07:00:49 GMT
Content-Type: text/html; charset=iso-8859-1

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<body onload="document.forms[0].submit()">

<noscript>
<p>
<strong>Note:</strong> Since your browser does not support JavaScript, you must
press the Continue button once to proceed.
</p>
</noscript>

<form action="https://IdentityProvider.com/SAML/SLO/Response" method="post">
<div>
<input type="hidden" name="RelayState"
value="0043bfc1bc45110dae17004005b13a2b"/>
<input type="hidden" name="SAMLResponse"
value="PHNhbWxwOkxvZ291dFJlc3BvbnNlIHhtbG5zOnNhbWxwPSJ1cm46b2FzaXM6bmFt
```

```
ZXM6dGM6U0FNTDoyLjA6cHJvdG9jb2wiIHhtbG5zPSJ1cm46b2FzaXM6bmFtZXM6
dGM6U0FNTDoyLjA6YXNzZXJ0aW9uIg0KICAgIElEPSJiMDczMGQyMWI2MjgxMTBk
OGI3ZTAwNDAwNWIxM2EyYiIgSW5SZXNwb25zZVRvPSJkMmI3YzM4OGNlYzM2ZmE3
YzM5YzI4ZmQyOTg2NDRhOCINCiAgICBJc3N1ZUluc3RhbnQ9IjIwMDQtMDEtMjFU
MTk6MDA6NDlaIiBWZXJzaW9uPSIyLjAiPg0KICAgIDxJc3N1ZXI+aHR0cHM6Ly9T
ZXJ2aWNlUHJvdmlkZXIuY29tL1NBTUw8L0lzc3Vlcj4NCiAgICA8c2FtbHA6U3Rh
dHVzPg0KICAgICAgICA8c2FtbHA6U3RhdHVzQ29kZSBWYWx1ZT0idXJuOm9hc2lz
Om5hbWVzOnRjOlNBTUw6Mi4wOnN0YXR1czpzdWNjZXNzIi8+DQogICAgPC9zYW1s
cDpTdGF0dXM+DQo8L3NhbWxwOkxvZ291dFJlc3BvbnNlPg==" />
</div>
<noscript>
<div>
<input type="submit" value="Continue"/>
</div>
</noscript>
</form>
</body>
</html>
```

10.2.6 HTTP Artifact绑定

在 HTTP Artifact 绑定中，SAML 请求、SAML 响应，或者这两者都使用一个称为凭证小替代物的参考来传输。一个单独的、同步的绑定，如 SAML SOAP 绑定，使用第 8 节中定义的凭证解析协议来为实际的协议消息互换凭证。

本绑定可以由 HTTP 重定向绑定（参见第 10.2.4 节）和 HTTP POST 绑定（参见第 10.2.5 节）组成，使用两个不同的绑定在一个单独的协议交换中传输请求和响应消息。

10.2.6.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.6.2 概述

HTTP Artifact 绑定的目的是为处理 SAML 请求端和响应端需要使用 HTTP 用户代理作为中间媒介进行通信，而中间媒介的限制又阻碍了整个消息（或消息交换）通过其传输的情况。这种情况可能出于技术原因或由于不愿将消息内容呈现给中间媒介（以及如果加密的使用是不实际的）。

由于随后解析凭证需要使用另一个同步的绑定，如 SOAP，因此在 SAML 消息发送端和接收端之间必须存在一条直接的通信路径，其方向与凭证的传输相反（消息和凭证的接收方必须能够发送一条<samlp:ArtifactResolve>请求返回给凭证的发送方）。凭证的发送方也必须在凭证待决过程中保持状态，这对载荷平衡的环境有意义。

10.2.6.3 消息编码

本绑定使用的凭证有两种方法进行编码。一种是将凭证编码成 URL 参数，另一种是将凭证置入 HTML 表单控件中。使用 URL 编码时，采用 HTTP GET 方式递送消息，而当使用表单编码时则采用 POST 方式。所有支持本绑定的端点必须支持这两种技术。

10.2.6.3.1 RelayState

本绑定传输的 SAML 凭证可以包含 RelayState 数据。值在长度上不能超过 80 个字节，并且应该独立于其他在消息传输期间可能存在或可能不存在的保护，由创建消息的实体提供完整性保护。鉴于空间的限制，签字是不可行的，但由于该值暴露于第三方的干扰中，因此实体应该采用校验和，伪随机值或其他类似的手段来确保该值不被篡改。

如果一个代表 SAML 请求的凭证带有 RelayState 数据，则 SAML 响应端必须采用一个同样支持 RelayState 机制的绑定来返回其 SAML 协议响应，并且必须将其与凭证一起接收到的准确的 RelayState 置于响应中相应的 RelayState 参数中。

如果一个代表 SAML 请求的凭证没有包含该值，或 SAML 响应消息在没有对应请求的情况下生成，那么 SAML 响应端可以包含 RelayState 数据，交由接收端基于协议子集或各方之间的预先约定来进行解释。

10.2.6.3.2 URL编码

为将凭证编码成 URL，先对凭证值进行 URL-编码，然后放到一个查询字符串参数中，命名为 SAMLart。

如果 SAML 凭证带有“RelayState”值，则该数据必须进行 URL-编码并且放在另一个查询字符串参数中，称为 `RelayState`。

10.2.6.3.3 表单编码

SAML 凭证的表单编码是通过 W3C HTML 定义的表单将其置入一个隐含的表单控件命令中。HTML 文件必须支持 W3C XHTML。该表单控件必须被命名为 `SAMLart`.可以包含其他的表单控件命令或表征，但它们不能是接收方处理消息所必需的。

如果 SAML 凭证伴随有“RelayState”值，则它必须被放在另一个隐含表单控件命令中，称为 `RelayState`，与 SAML 消息的表单相同。

表单的 `action` 属性必须是接收方的 HTTP 端点，对使用本绑定的协议或协议子集来说，SAML 消息就传送到这个端点。method 属性必须是“POST”。

可以使用用户代理所支持的任何技术来服从格式，而任何支持这一点所需的表单内容都可以被包含，例如提交控制和客户端脚本命令。然而，接收端必须能够不考虑这种表单提交的机制来处理凭证。

任何包含的表单控件值必须被转换以便安全的包含在 XHTML 文件中。这包括转换诸如引号之类的字符成为 HTML 实体。

10.2.6.4 凭证格式

本绑定中的凭证是一个短而不透明的字符串。可以定义并使用不同的类型而不影响到绑定。重要的特性是凭证接收方鉴别凭证发送方的能力，对于干扰和伪造物的抵抗能力，唯一性和紧密性。

任何凭证的一般格式都包括两字节的强制凭证类型码以及两字节的索引值，标识发送方的凭证解析业务的特定端点，如下所示：

```
SAML_artifact     := B64(TypeCode EndpointIndex RemainingArtifact)
TypeCode          := Byte1Byte2
EndpointIndex     := Byte1Byte2
```

标记 B64（`TypeCode EndpointIndex RemainingArtifact`）代表对 `TypeCode`，`EndpointIndex`，和 `RemainingArtifact` 的连接串采用基 64（参见 IETF RFC 2045）的应用。

SAML 凭证的创建推荐采用如下过程：

- 对每个发送方指定一个标识URI，也称为发送方实体（或提供者）ID。参见第8节此类标识符的讨论。
- 发送方通过对标识URL采用SHA-1哈希算法构造凭证的`SourceID`构件。哈希值不编码成十六进制值。

  注 1 — NIST（国际标准和技术协会）现在鼓励使用 SHA-256（256 比特编码密钥的安全哈希算法）代替 SHA-1。
- `MessageHandle`值是由发送方产生的加密的强随机或伪随机数字序列（参见IETF RFC 1750）构建的。序列由至少16字节长的值组成。这些值如需要应该被填补到总长度为20字节。

  注 2（资料性的）— PE4（参见[OASIS Errata Document]）建议在上段的末尾增加以下文字：

  尽管一般的凭证结构与SAML较早版本的类似，并且下述的独立格式的类形码与之前版本定义的格式不冲突，但很明显，SAML 2.0凭证和较早所有规范中的是不一致的，并且不是为SAML2.0专门定义的凭证格式禁止在本绑定中使用。

以下描述了 SAML V2.0 定义的单一的凭证类型。

10.2.6.4.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:artifact-04

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.6.4.2 格式详细信息

SAML V2.0 定义了类型码为 0x0004 的凭证类型。该凭证类型定义如下：

```
TypeCode           := 0x0004
RemainingArtifact  := SourceID MessageHandle
SourceID           := 20-byte_sequence
MessageHandle      := 20-byte_sequence
```

SourceID 是一个 20 字节的序列，凭证接收方用它来确定凭证发送方的身份和可能的解析端点的集合。

假定目的站点会为相应的 SAML 响应端维护一个 SourceID 值和一个或多个编有索引的 URL 端点（或地址）的表格。第 9 节可以用来实现这个目的。一收到 SAML 凭证，接收方在发送一条 SAML<samlp:ArtifactResolve>消息之前，会先确定 SourceID 是否属于未知的凭证发送方，并且使用 EndpointIndex 获得 SAML 响应端的位置。

两个具有相同接收方的凭证签发者必须使用不同的 SourceID 值。MessageHandle 值的构建须遵循如下原则：它们不应该与发送站点引用消息的内容存在可预测的联系，而且构建或猜测一个合法的、独特的消息句柄值必须是不可实现的。

10.2.6.5 消息交换

使用本绑定进行的 SAML 会话所使用的系统模型是一个请求-响应模型，在该模型中凭证引用代替实际的消息内容，在 HTTP 响应中发送给用户代理，在 HTTP 请求中传送给消息接收方的。这些交换发生之前、之间以及之后的 HTTP 交互是隐含的。SAML 请求端和 SAML 响应端都假定是 HTTP 响应端。

另外，假定凭证接收方在经过用户代理接收凭证时，会使用本建议书定义的凭证解析协议，与凭证签发方进行一个独立的、直接的交换。该交换必须使用一个不用 HTTP 用户代理作为中介的绑定，例如 SOAP 绑定。一旦成功获得 SAML 协议消息，凭证被丢弃，初始的 SAML 协议交换的处理继续执行（或终止，如果消息是一个响应的话）。

签发和传送一个凭证，及其后的解析步骤，构成了整个 SAML 协议交换的一半。本绑定能被用来传送 SAML 协议交换的任意一半或这两者。与其对应的绑定，例如 HTTP 重定向（参见第 10.2.4 节）或 POST（参见第 10.2.5 节）绑定，可以被用来承载交换的另一半。以下流程假定使用凭证映射来传送两者。下面的图 10-4 描绘了消息交换过程。

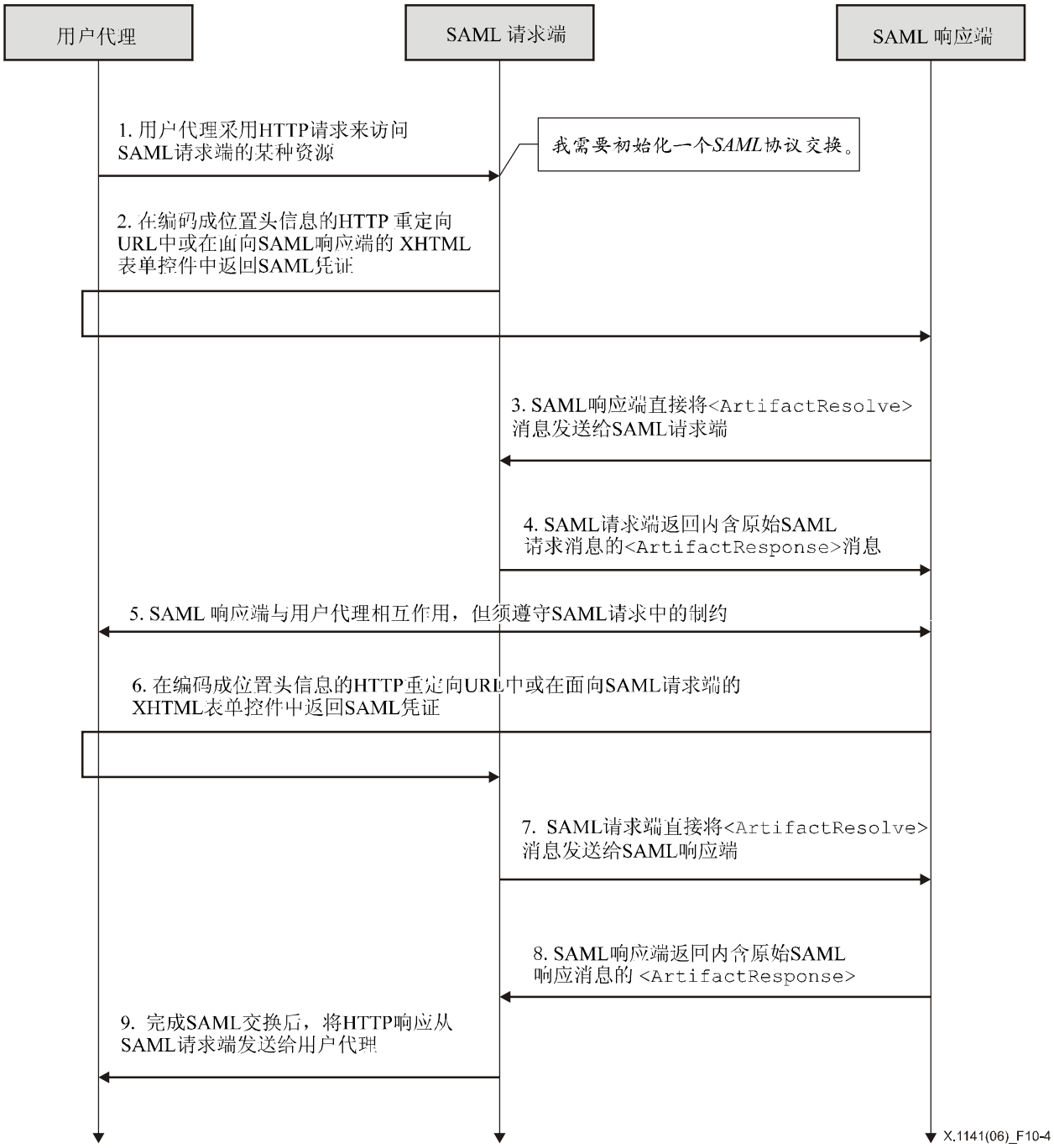

图 10-4/X.1141－HTTP Artifact 消息交换

1） 首先，用户代理对系统实体做出任意的HTTP请求。在处理该请求的过程中，系统实体决定启动一次SAML协议交换。

2） 系统实体作为SAML请求端对用户代理提出的HTTP请求返回一个代表SAML请求的凭证作为应答。

- 如果是URL编码形式，该凭证被编码成HTTP响应的Location头返回，其HTTP状态必须设为303或302。SAML请求端可以在HTTP响应中包含其他的内容和表达以便用户代理的消息传输，如IETF RFC 2616中定义的那样。用户代理通过发布一个HTTP GET请求给SAML响应端来传输凭证。
- 如果是格式编码形式，则该凭证在一个XHTML文件中返回，文件中包含10.2.6.3.3定义的表单和内容。用户代理通过发布一个HTTP POST请求给SAML响应端来传输凭证。

3） SAML响应端通过检查凭证（准确过程取决于凭证类型）来确定SAML请求端，并使用一个直接的SAML绑定发送一个包含凭证的<samlp:ArtifactResolve>请求给SAML请求端，临时转换角色。

4） 假定符合必要的条件，SAML请求端将返回一个包含它希望SAML响应端处理的起始SAML请求消息的<samlp:ArtifactResponse>。

5） 一般说来，SAML响应端可以立即返回一条SAML凭证给SAML请求作为应答，也可以返回任意的内容以便与完成请求所必需的用户代理进行后续交互。特定的协议和协议子集可能包含请求端容许此类交互的主动性程度的指示机制（举例来说，<samlp:AuthnRequest>中的IsPassive属性）。

6） 最终响应端应该返回一条SAML凭证给用户代理，以返回给SAML请求端。SAML响应凭证与步骤2中所述的SAML请求凭证的方式一样返回。

7） 与步骤3中类似，SAML请求端通过检查凭证来确定SAML响应端，并使用一个直接的SAML绑定发送一个包含凭证的<samlp:ArtifactResolve>请求给SAML响应端。

注（资料性的）— PE31 （参见 OASIS PE:2006） 建议将步骤 6 中的最后一句替换为：

与步骤 3 中类似，SAML 请求端通过检查凭证来确定 SAML 响应端，并使用一个同步的 SAML 绑定签发一个包含凭证的<samlp:ArtifactResolve>请求给 SAML 响应端。

8） 与步骤4中类似，假定符合必要的条件，SAML响应端将返回一个包含它希望请求端处理的SAML响应消息的<samlp:ArtifactResponse>。

9） 一收到SAML响应，SAML请求端即返回任意的一条HTTP响应给用户代理。

10.2.6.5.1 HTTP及缓存考虑

HTTP 代理和用户代理中介不应缓存 SAML 协议消息。为确保这一点，以下规则应该遵守：

当使用 HTTP1.1 返回 SAML 凭证时，HTTP 响应端应该：

- 包含一个Cache-Control头信息字段，设置为“no-cache, no-store”。
- 包含一个Progma指示头信息字段，设置为“no-cache”。

对于 HTTP 头的使用没有其他限制。

10.2.6.5.2 安全性考虑

本绑定使用了间接传输消息引用，随后通过直接的交换来返回实际消息的组合。结果，消息参考（凭证）自身不需要被认证或完整性保护，但返回实际消息的回收请求/响应交换可以相互认证以及完整性保护，取决于使用环境而定。

如果实际的 SAML 协议消息是针对特定接收方的，则凭证的发行方必须在返回实际消息前认证寄送方随后的<samlp:ArtifactResolve>消息。

凭证发送到和接收自用户代理的过程中必须带有信任度保护，或者应该使用 TLS 1.0。返回实际消息的回收请求/响应交换可以被保护，取决于使用环境而定。

总之，本绑定依赖于凭证作为一个难于仿照的短期的引用，并且对返回实际消息的回收请求/响应采用其他的安全措施。凭证发行方强制所有的凭证必须具有一个单一用法的语义。

进一步的，建议凭证接收方也给其接收到的凭证值强制规定一个单一用法的语义，以此来防止攻击者通过用户代理干预凭证的解析并将其重新交给凭证接收方。如果解析凭证的尝试不完全成功，则该凭证应该被放进一张被阻止凭证列表中一段时间，这段时间应超过凭证发行方解析凭证的可接受的合理时间范围。

没有定义一种机制来保护凭证和如果存在的“RelayState”值之间关系的完整性。也就是说，攻击者可能通过转换与每个凭证相关联的“RelayState”值来重组一对有效的 HTTP 响应。结果，“RelayState”信息的产生者/消费者必须注意不要不加以额外的预防措施（例如基于通过凭证获取的 SAML 协议消息中的信息）就将敏感的状态信息与“RelayState”值相结合。

10.2.6.6 错误报告

拒绝与 SAML 请求端执行消息交换的 SAML 响应端应该返回一个带有 `urn:oasis:names:tc:SAML:2.0:status:RequestDenied` 的第二级`<samlp:StatusCode>`值的响应消息。

消息交换期间的 HTTP 交互禁止使用 HTTP 错误状态码来指示 SAML 处理中的失败，因为用户代理对于 SAML 协议交换来说不是完整的一方。

如果凭证发行方收到一条它能理解的`<samlp:ArtifactResolve>`消息，它必须返回一条带有 `urn:oasis:names:tc:SAML:2.0:status:Success` 的`<samlp:StatusCode>`值的`<samlp:ArtifactResponse>`，即使它不返回相应的消息（举例来说，凭证请求端未被授权来接收消息或凭证不再有效）。

10.2.6.7 元数据方面的考虑

对于 HTTP 凭证映射的支持应该反映为指明 URL 端点，在那里应该发送针对特定协议或协议子集的请求和响应。也可提供单独的端点或不同的请求和响应端点。一个或多个编有索引的用来处理`<samlp:ArtifactResolve>`消息的端点也应该被描述。

10.2.6.8 使用HTTP Artifact的SAML消息交换举例

本例中，一个`<LogoutRequest>`和 `<LogoutResponse>`消息对使用 HTTP Artifact 绑定进行交换，凭证的解析使用映射到 HTTP 的 SOAP 映射实现。

首先是将要交换的实际的 SAML 协议消息：

```
<samlp:LogoutRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="d2b7c388cec36fa7c39c28fd298644a8"
IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://IdentityProvider.com/SAML</Issuer>
    <NameID
Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">005a06e0-ad82
-110d-a556-004005b13a2b</NameID>
    <samlp:SessionIndex>1</samlp:SessionIndex>
</samlp:LogoutRequest>

<samlp:LogoutResponse xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="b0730d21b628110d8b7e004005b13a2b"
InResponseTo="d2b7c388cec36fa7c39c28fd298644a8"
    IssueInstant="2004-01-21T19:00:49Z" Version="2.0">
    <Issuer>https://ServiceProvider.com/SAML</Issuer>
    <samlp:Status>
        <samlp:StatusCode
Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
    </samlp:Status>
</samlp:LogoutResponse>
```

本绑定不对步骤 1 中从用户代理处发出的初始 HTTP 请求进行定义。为发起注销协议交换，SAML 请求端返回以下 HTTP 响应，包含一个 SAML 凭证。填入下面的 HTTP Location 头中的那一行是文档格式化的结果，实际的头中不填入数据值。

```
HTTP/1.1 302 Object Moved
Date: 21 Jan 2004 07:00:49 GMT
Location:
https://ServiceProvider.com/SAML/SLO/Browser?SAMLart=AAQAADWNEw5VT47wcO4zX%
2FiEzMmFQvGknDfws2ZtqSGdkNSbsW1cmVR0bzU%3D&RelayState=0043bfc1bc45110dae170
04005b13a2b
Content-Type: text/html; charset=iso-8859-1
```

SAML 响应端随后在步骤 3 和 4 中，使用凭证解析协议和 SOAP 映射将其接收到的凭证解析成实际的 SAML 请求，如下所述：

步骤 3：

```
POST /SAML/Artifact/Resolve HTTP/1.1
Host: IdentityProvider.com
Content-Type: text/xml
Content-Length: nnn
SOAPAction: http://www.oasis-open.org/committees/security
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Body>
        <samlp:ArtifactResolve
             xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
             xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
             ID="_6c3a4f8b9c2d" Version="2.0"
             IssueInstant="2004-01-21T19:00:49Z">
             <Issuer>https://ServiceProvider.com/SAML</Issuer>
             <Artifact>
             AAQAADWNEw5VT47wcO4zX/iEzMmFQvGknDfws2ZtqSGdkNSbsW1cmVR0bzU=
             </Artifact>
        </samlp:ArtifactResolve>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

步骤 4：

```
HTTP/1.1 200 OK
Date: 21 Jan 2004 07:00:49 GMT
Content-Type: text/xml
Content-Length: nnnn

<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Body>
        <samlp:ArtifactResponse
             xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
             xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
             ID="_FQvGknDfws2Z" Version="2.0"
             InResponseTo="_6c3a4f8b9c2d"
             IssueInstant="2004-01-21T19:00:49Z">
             <Issuer>https://IdentityProvider.com/SAML</Issuer>
             <samlp:Status>
                    <samlp:StatusCode
             Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
             </samlp:Status>
             <samlp:LogoutRequest ID="d2b7c388cec36fa7c39c28fd298644a8"
                    IssueInstant="2004-01-21T19:00:49Z"
                    Version="2.0">
                    <Issuer>https://IdentityProvider.com/SAML</Issuer>
                    <NameID
Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">005a06e0-ad82
-110d-a556-004005b13a2b</NameID>
                    <samlp:SessionIndex>1</samlp:SessionIndex>
             </samlp:LogoutRequest>
        </samlp:ArtifactResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

在任何可能的不特定的交互发生之后，SAML 响应端在步骤 6 中返回包含在 HTTP 响应中的第二个 SAML 凭证：

```
HTTP/1.1 302 Object Moved
Date: 21 Jan 2004 07:05:49 GMT
Location:
https://IdentityProvider.com/SAML/SLO/Response?SAMLart=AAQAAFGIZXv5%2BQaBaE
5qYurHWJO1nAgLAsqfnyiDHIggbFU0mlSGFTyQiPc%3D&RelayState=0043bfc1bc45110dae1
7004005b13a2b
Content-Type: text/html; charset=iso-8859-1
```

SAML 响应端随后在步骤 7 和 8 中，使用凭证解析协议和 SOAP 映射将其接收到的凭证解析成实际的 SAML 请求，如下所述：

步骤 7：

```
POST /SAML/Artifact/Resolve HTTP/1.1
Host: ServiceProvider.com
Content-Type: text/xml
Content-Length: nnn
SOAPAction: http://www.oasis-open.org/committees/security
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Body>
        <samlp:ArtifactResolve
             xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
             xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
             ID="_ec36fa7c39" Version="2.0"
             IssueInstant="2004-01-21T19:05:49Z">
             <Issuer>https://IdentityProvider.com/SAML</Issuer>
             <Artifact>
             AAQAAFGIZXv5+QaBaE5qYurHWJO1nAgLAsqfnyiDHIggbFU0mlSGFTyQiPc=
             </Artifact>
        </samlp:ArtifactResolve>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

步骤 8：

```
HTTP/1.1 200 OK
Date: 21 Jan 2004 07:05:49 GMT
Content-Type: text/xml
Content-Length: nnnn

<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Body>
        <samlp:ArtifactResponse
             xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
             xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
             ID="_FQvGknDfws2Z" Version="2.0"
             InResponseTo="_ec36fa7c39"
             IssueInstant="2004-01-21T19:05:49Z">
             <Issuer>https://ServiceProvider.com/SAML</Issuer>
             <samlp:Status>
                    <samlp:StatusCode
             Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
             </samlp:Status>
             <samlp:LogoutResponse ID="_b0730d21b628110d8b7e004005b13a2b"
                    InResponseTo="_d2b7c388cec36fa7c39c28fd298644a8"
                    IssueInstant="2004-01-21T19:05:49Z"
                    Version="2.0">
                    <Issuer>https://ServiceProvider.com/SAML</Issuer>
                    <samlp:Status>
                           <samlp:StatusCode
             Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
                    </samlp:Status>
             </samlp:LogoutResponse>
        </samlp:ArtifactResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

10.2.7 SAML URL绑定

URI 是参考某个资源的协议无关的方法。该映射不是一个简单的 SAML 请求/应答映射，它还支持用一个单独的<saml:AssertionIDRef>将<samlp:AssertionIDRequest>消息封装成为一个 URI 的变体。一个成功的请求的结果是一个 SAML<saml:Assertion>元素（但不是一个完整的 SAML 应答）。

与 SOAP 类似，URI 变体能够在多种底层传输之上存在。本绑定具有传输独立的性质，但带有 TLS1.0 的 HTTP 的使用是必需的（必须实现）。

注（资料性的）— PE24（参见OASIS PE:2006）建议将上段替换为下列文字：

与SOAP类似，URI变体能够在多种底层传输之上存在。本绑定具有传输独立的性质，但必须实现HTTP URI。

10.2.7.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:bindings:URI

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

10.2.7.2 SAML URI绑定的协议独立性

以下各节定义 SAML URI 绑定独立于 URI 变体处理的底层传输协议的方面。

一个 SAML URI 引用标识一个特定的 SAML 断言。解析这个 URI 的结果必须是一个包含该断言的消息，或者是一个特定的传输错误。消息的特定格式取决于底层传输协议。如果传输协议允许返回内容被说明，如 HTTP1.1，那么断言可以以任何允许的格式进行编码。否则，断言必须以某种表单返回，该表但能够被清楚地翻译为或转换成断言的 XML 序列。

如果将来要解析相同的 URI 引用，必须在这种情况下才返回同样的 SAML 断言或错误。也就是说，引用可以是持续的但如果引用断言，则必须一致性的引用同样的断言。

10.2.7.3 安全性考虑

对 SAML 断言的间接使用阻止了风险，如果对于结果引用的映射是不安全的。特殊的威胁及其严重性取决于断言的使用。总之，解析一个 URI 引用成为 SAML 断言的结果只应该在如下条件下被信任，即请求端能够确信响应端的身份并且传输中内容没有被改变。

对于断言本身进行签字常常是不够的，因为 URI 引用从本质来说是对于请求端在某种程度上是不透明的。请求端应该具有独立的手段来确保返回的断言确实是由 URI 代表的断言；这一点通过对响应端的认证和响应完整性的信任来完成。

10.2.7.4 MIME封装

对于支持 MIME 作为内容说明和封装机制的解析协议，结果断言应该作为一个 MIME 实体返回，其类型为 application/samlassertion+xml，如附录二中定义。

10.2.7.5 HTTP URI的使用

声称具有 SAML URI 绑定一致性的 SAML 权威必须实现对于 HTTP 的支持。本节描述使用 HTTP URI 的一些规定，包括 URI 句法，HTTP 头，和错误报告。

10.2.7.5.1 URI 句法

总的来说，只要是在对参考负责的 SAML 权威创建的消息中包含，那么可允许的 SAML URI 参考的句法不受限制。然而，权威必须支持一个 URL 端点，在这里能发送一个带有一个单独的名为 ID 的查询字符串参数的 HTTP 请求。独立于此参数之外的端点 URL 自身中必须没有查询字符串。

举个例子，如果一个权威的引证端点是“https://saml.example.edu/assertions”，则对于一个带有 ID 为 abcde 的断言的请求可以发送到：

```
https://saml.example.edu/assertions?ID=abcde
```

此类 ID 查询不允许使用通配符。

注（资料性的）— PE31（参见OASIS PE:2006）建议将上段文字替换为：

注意URI句法不支持在此类查询中使用通配符。

10.2.7.5.2 HTTP及缓存考虑

HTTP 代理不应缓存 SAML 断言。为确保这一点，以下规则应该遵守：

当使用 HTTP1.1 返回 SAML 断言时，HTTP 响应端应该：

- 包含一个缓存控制头信息字段，设置为“no-cache, no-store”。
- 包含一个编译指示头信息字段，设置为“no-cache”。

10.2.7.5.3 安全性考虑

IETF RFC 2617 描述了在 HTTP 环境中使用基础验证或消息摘要验证方案时可能遭受的攻击。

强烈建议使用 TLS 1.0 作为认证、完整性保护和信任度的方法。

10.2.7.5.4 错误报告

作为一个 HTTP 协议交换，适当的 HTTP 状态码应该被用来指明请求结果。举例来说，拒绝与 SAML 请求端执行消息交换的 SAML 响应端应该返回一个“403 Forbidden”应答。如果响应端不认识指定的断言，那么应该返回一条“404 未发现”响应。在这些情况下，HTTP 消息体的内容不再有效。

10.2.7.5.5 元数据考虑

对于 HTTP 上的 URI 绑定的支持反映为指明一个 URL 端点，在那里对于任意断言的请求等待发送。

10.2.7.5.6 使用HTTP URI的SAML消息交换举例

以下是请求一个断言的例子。

```
GET /SamlService?ID=abcde HTTP/1.1
Host: www.example.com
```

以下是提供必需的断言的相应响应的例子。

```
HTTP/1.1 200 OK
Content-Type: application/samlassertion+xmlCache-Control: no-cache, no-store
Pragma: no-cache
Content-Length: nnnn

<saml:Assertion ID="abcde" ...>
...
</saml:Assertion>
```

11 SAML协议子集

本节具体规定用于确定 SAML 断言和请求/响应消息在通信协议中如何应用的协议子集与框架，并具体规定用于确定 SAML 属性值句法和命名约定的协议子集。

11.1 协议子集的概念

有一种类型的 SAML 协议子集讲述的一些规则是用来描述如何从一个框架或者协议中将 SAML 断言(断言）嵌入和提取。这种协议子集描述了 SAML 断言（断言）是如何通过发起者、发起者与接收端之间的通信以及随后在目的地的处理与其他的实体（比如各种类型的文件、通信协议的数据元素等）结合在一起的。用于将 SAML 断言嵌入到特定类型的实体或者从特定<FOO>实体中剥离 SAML 断言的一系列规则就形成了 SAML<FOO>规范。

比如，SAML 的 SOAP 协议子集描述了 SAML 断言如何被添加到 SOAP 消息中、SAML 断言如何影响 SOAP 标题头以及 SAML 的相关错误状态应该如何在 SOAP 消息中体现。

另一种类型的 SAML 协议子集针对特定的应用环境和前后关系定义了一系列关于常规 SAML 协议和断言性能的使用规范。这种类型的协议子集可能会约束可选择性、注重特定 SAML 功能性（比如特性、环境或者附属性能）的效用、并且从其他方面定义了协议子集参与者需要遵守的处理规则。

后者的一个典型的例子就是那些 SAML 属性的定位。在属性命名、值句法、以及通过 XML 属性的应用的包含元数据。协议子集定义了怎样使用这些原理能够比第 8 节定义的普通规则具有更好特性，依附协议子集保证的适应性约束能够带来互操作性。

当处理特定类型的属性消息时或者与要求较强严密性的扩展系统或者其他开放标准进行交互时，属性协议子集为约束 SAML 属性表示提供了必需的定义。

本建议书的目的是足够详细的选列出一系列不同种类的协议子集，这些协议子集可以保证独立的执行工具可以互操作。

11.2 附加协议子集规范

本建议书选列了一些协议子集定义，并且其他的在将来可能会被扩展。下面段落给出了规定协议子集的指南。

11.2.1 指定的协议子集的指南

本部分提供了一个需要每种协议子集来解决的问题的列表。

1） 规定一个URI来为作者唯一的识别协议子集、邮政或者电子的联系信息，并且提供以前定义的协议子集作参考来分析新协议子集是更新了还是过时了。

2） 描述协议子集成员之间的系列交互活动。成员所使用软件的约束和每种交互活动涉及的协议必须被明确的给出。

3） 标识每种交互活动涉及的成员，包括涉及多少成员以及是否涉及中间代理。

4） 规定每种交互活动所涉及放认证方法，包括认证是否需要以及可接受的认证方法。

5） 标识对消息完整性支持的等级，包括用来确保消息完整性的机制。

6） 标识对机密性支持的等级，包括第三方是否能够查看SAML消息和断言的内容，协议子集是否需要机密性，以及为了达到机密性而推荐的机制。

7） 标识错误的状态，包括每个参与者的错误状态，尤其是那些收到和传递SAML断言或消息的那些。

8） 标识安全的考虑事项，包括对威胁的分析以及对策的描述。

9） 标识协议子集定义和使用的SAML确认方法标识符。

10）标识协议子集定义和/或利用的相关SAML元数据。

11.2.2 指定的属性协议子集的指南

本部分提供了一个必须特别的被属性协议子集处理的项目列表。

1） 规定一个URI来为作者唯一的识别协议子集、邮政或者电子的联系信息，并且提供新规范更新或废止了的以前定义的协议子集作参考来分析。

2） SAML<Attribute>元素的命名格式和命名特性的可接受值的句法和约束。

3） SAML <Attribute>元素中可以使用的协议子集所定义的任何额外的符合XML属性。

4） 决定协议子集所定义的SAML<Attribute>元素的等同性，如在处理属性、查询等时的使用。

5） SAML<AttributeValue>元素中可接受值的句法和约束，包括各种XML特性是否能够或者应当被使用。

11.3 确认方法标识符

第 8 节将<SubjectConfirmation>元素定义为一个附加可选<SubjectConfirmationData>的Method。在特定协议子集的关联中，<SubjectConfirmation>元素被信任方用来用于确认来自于一个系统实体的请求或者消息，该系统实体与断言的主体相关联。

Method 属性指示信任方用来做出该决定的特定方法。这可能与先前执行的认证有或者没有任何联系。与验证的关联不同，在<SubjectConfirmationData>元素之中，主体的确认方法通常附带着附加信息，比如证书或者密钥，这将允许信任方做出必要的验证。还定义了一些用来约束查证发生的条件。

正如所预期的，协议子集为<ConfirmationMethod>定义和使用了几个不同的值，每个值对应于一个不同的 SAML 使用场景。后面的方法是为本建议书中规定的协议子集和其他认为这些方法有用的协议子集的使用而定义的。

11.3.1 密钥持有者

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

<SubjectConfirmationData>元素之中必须有一个或者更多的<ds:KeyInfo>元素。一个名为 xsi:type 的属性可能会出现在<SubjectConfirmationData>元素之中，如果出现的话，必须被设在**saml:KeyInfoConfirmationDataType**（命名空间的前缀是任意的，但必须参照 SAML 声名的命名空间）。

正如 W3C Signature 描述的那样，每个<ds:KeyInfo>元素持有一个密钥或者信息来使应用程序能够获得一个密钥。一个特定密钥的持有者被认为是断言方做出断言的对象。

与 W3C Signature 保持一致，每个<ds:KeyInfo>元素必须识别一个独特的密钥。多种密钥可以被不同的<ds:KeyInfo>元素识别，比如当不同的信任方需要不同的确认密钥时。

例如：名为"By-Tor"的密钥持有者或者是名为"Snow Dog"的密钥持有者作为对象本身可以批准它自己。

```
<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>
```

11.3.2 发送端担保

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

需要明确的是，没有什么信息是对断言的使用关联有效。信任方应使用其他的方法来对断言作进一步的查证处理，在确认时会利用到可能会在<SubjectConfirmationData>元素中出现的属性来使它服从可选约束。

11.3.3 承载者

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

断言的主体是断言的承载者，如第 8 节定义的那样，使用可能在<SubjectConfirmationData>元素中出现的属性来使它服从可选约束。

例子：作为对 ID"_1234567890"请求的应答，在 2004 年 3 月 19 日 1:37 PMGMT 之前，倘若断言被在消息中发送给"https://www.serviceprovider.com/saml/consumer"，作为对象的断言传送者可以批准它自己本身。

```
<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>
```

11.4 SAML的SSO协议子集

为支持浏览器和其他客户端设备的单点登录（SSO）规定了一系列协议子集。

- 为支持web单点登录，第8节中认证请求协议规定基于网络浏览器的协议子集。
- 为支持增强型客户端，规定附加的web SSO协议子集。
- 第8节对前向信道（浏览器）和备份信道绑定定义了单点注销和命名标识符管理协议的协议子集。
- 使用cookies为身份提供者发现定义附加的协议子集。

11.4.1 Web浏览器SSO协议子集

在 web 浏览器 SSO 协议子集所支持的情境之中，一个网络用户不但可以访问一个服务提供商的资源，而且可以访问身份提供者，这样服务提供商和想要的资源都被理解或者接受。网络用户由身份提供者鉴别（或者已经鉴别），然后身份提供者发布一个认证断言（可能包含服务提供商信息），而服务提供商使用这个认证断言来为网络用户建立安全关联。在这个过程中，服务提供商和身份提供者之间还会为用户建了一个名称认证，使它服从用户交互和应答的参数。

为了实现这一情景，一种 SAML 验证请求协议的协议子集与 HTTP Redirect、HTTP POST 和 HTTP Artifact bindings 联合在一起使用。

假设用户使用一个标准商业浏览器并且可以通过一些 SAML 范围之外的方法从身份提供者获得认证。

11.4.1.1 必需的信息

鉴定：urn:oasis:names:tc:SAML:2.0:profiles:SSO:浏览器

联系方式：security-services-comment@lists.oasis-open.org

SAML 确认方法标识符：（这个协议子集使用 SAML V2.0 "承载者"确认方法标识符，）urn:oasis:names:tc:SAML:2.0:cm: 承载者

说明：见下文。

更新：无。

11.4.1.2 协议子集概述

图 11-1 图示阐明了实现 SSO 的基本模版。该协议子集描述了以下步骤。在单独的步骤之中，根据用于该步骤和其他与实施相关的行为的绑定，可能会有一个或者多个实际消息交换。

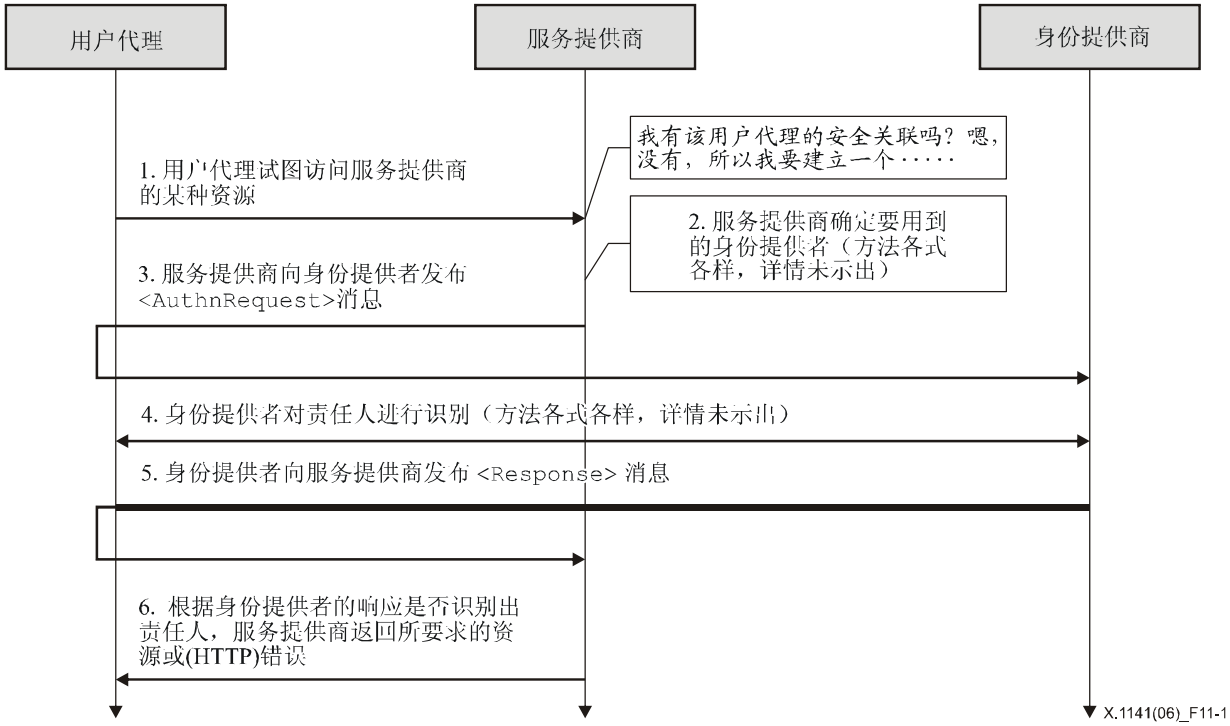

图 11-1/X.1141－实现SSO的基本模版

1） 服务提供商的HTTP请求

在第1步中，通过一个HTTP用户代理，责任人向不具备安全关联的服务提供商发送一个针对受保护资源的HTTP请求。

2） 服务提供商决定身份提供者

在第2步中，服务提供商得到适合于认证请求协议-支持先前映射-的认证提供者的最终位置。完成这一步的方法是被动执行的。服务提供商可以使用第8.7.4节部分描述的SAML认证提供者（身份提供者）发现规范。

3） 服务提供商向身份提供者发布<AuthnRequest>请求

在第3步中，服务提供商发布一个<AuthnRequest>请求消息，由用户代理转发给身份提供者。通过用户代理，HTTP Redirect, HTTP POST 或 HTTP Artifact binding都可以用来转发消息给身份提供者。

4） 身份提供者验证起始发起者

在第4步中，身份提供者通过本规范以外的其他方法验证责任人。这可能需要一个新的认证行为或者重用一个已存在的认证方法。

5） 身份提供者发布<Response>应答给服务提供商

在第5步中，身份提供者发布一个<Response>消息，由用户代理转发给服务提供商。通过用户代理，HTTP POST或者HTTP Artifact binding都可以用来转发消息给服务提供商。这一消息中可以包含认证错误或者认证断言。不能使用HTTP Redirect binding，因为应答消息的长度会超过多数用户代理允许的URL长度。

6） 服务提供商接受或者拒绝责任人的访问

在第6步中，从身份提供者收到应答之后，服务提供商可以对责任人用户代理自身的认证错误做出应答，或者为责任人设置（访问的）安全关联并返回被请求的资源。

身份提供者可以脱离前述步骤而在第5步发起这个协议子集行为，并发布一个<Response>消息给服务提供商。

11.4.1.3 协议子集描述

如果协议子集是由服务提供商发起的，就从第 11.4.1.3.1 节开始。如果协议子集是由身份提供者发起的，则从第 11.4.1.3.5 节开始。下面要讲述的是：

单点登录（SSO）服务

这是认证请求协议在身份提供者处的端点，用户代理将<AuthnRequest>消息（或者表示此消息的凭证）转发到身份提供者。

断言使用者服务

这是认证请求协议在服务提供商处的端点，用户代理将<Response>应答消息（或者表示此消息的凭证）转发到服务提供商。

11.4.1.3.1 对服务提供商的HTTP请求

如果对服务提供商的访问是首次访问，那么对资源的强制请求可以初始协议子集。对请求的表单没有限制。服务提供商可以自由地使用它所希望的任何方法来将随后的交互与初始请求结合起来。每一种绑定提供一个 RelayState 机制，服务提供商可以使用这个机制来将协议子集交互与源请求结合起来。服务提供商应该在 RelayState 属性中暴露尽可能少的请求，除非使用的协议子集没有保密尺度的需要。

11.4.1.3.2 服务提供商确定身份提供者

这一步是从属执行。服务提供商可以使用第 11.4.3 节中描述的 SAML 身份提供者发现规范。服务提供商也可以选择去重定向用户代理到另一个能够确定适当身份提供者的服务提供商那里。在这种情况中，服务提供商可以发布一个<AuthnRequest>请求（见下一步）给要接替确定身份提供者的服务提供商，或者它从自身利益出发可能依靠一个中间服务者来发布这个<AuthnRequest>请求消息。

11.4.1.3.3 服务提供商发布<AuthnRequest>请求给身份提供者

一旦基于服务提供商为发送<AuthnRequest>请求而选择的SAML绑定，选定了一个身份提供者，那么SSO单点登录服务的位置就确定了。为此目的还可能使用元数据。在用户代理对 HTTP 请求的应答之中，一个包含了 <AuthnRequest> 消息或者其重建消息的 HTTP 应答依靠所使用的 SAML 绑定而被转发到身份提供者的 SSO 单点登录服务上。

发往 SSO 单点登录服务的 HTTP 应答和并发的 HTTP 请求的准确格式是由使用的 SAML 绑定定义的。在第 11.4.1.4.1 节中讲述了<AuthnRequest>消息的内容应符合的协议子集特殊规则。如果使用了 HTTP Redirect or POST binding，在这一步中<AuthnRequest>消息就被直接转发给身份提供者。如果使用了 HTTP Artifact binding，身份提供者使用了第 11.4.6 节中定义的 Artifact 决定规范，Artifact 决定规范使用比如 SOAP 映射产生一个复查消息给服务提供商来重新获得<AuthnRequest>消息。

建议的做法是使用安全传输层协议（TLS1.0）承载 HTTP 交换以保证机密性和消息完整性。如果需要请求发起者的认证，<AuthnRequest>消息可能会被签字。如果使用了 HTTP Artifact binding，当重建被摒弃时，将提供一个认证请求发起者的备用方法。

正如本建议书中所描述的，身份提供者必须处理<AuthnRequest>消息。这将把随后的交互与用户代理约束在一起，比如，如果包括被动特性时。

11.4.1.3.4 身份提供者标识责任人

在前一步中或者随后的某个时候，身份提供者必须建立了责任人的身份认证（除非它返回了一个错误消息给服务提供商）。强制请求<AuthnRequest>特性的值如果是 true，则身份提供者有责任从新建立这个身份认证，而不能仍依靠责任人可能已具有的身份状态。另外在其他方面，身份提供者可以使用任何方法去认证用户代理，依照以<RequestedAuthnContext>元素的形式出现在<AuthnRequest>请求中的任何要求。

11.4.1.3.5 身份提供者发布<Response>应答给服务提供商

不管<AuthnRequest>请求是成功还是失败，身份提供者都需要制作一个包含一个<Response>消息或者一个凭证 的 HTTP 应答给用户代理，基于所使用的 SAML 绑定转发给服务提供商的断言使用者服务（简称 ACS）。

所使用的 SAML 绑定定义了这个 HTTP 应答和随后对断言使用者服务（ACS）的 HTTP 请求的准确格式。第 11.4.1.4.2 节中包括了<Response>消息内容的协议子集特殊规则。如果使用了 HTTP POST binding，<Response>消息将在这一步被直接转发给服务提供商。如果使用了 HTTP Artifact binding，服务提供商使用第 11.4.6 节中定义的 Artifact 决定规范，它使用比如 SOAP 映射产生一个针对身份提供者的复查信号来重新获得<Response>消息。

断言使用者服务（ACS）的位置可能决定使用元数据。身份提供者必定有一些方法设置成如此：这个位置其实是由服务提供商控制的。服务提供商可能会指定 SAML binding 和特定的断言使用者服务（ACS）在它们的<AuthnRequest>消息中使用并且身份提供者必定尽可能的遵守它。

建议的做法是使用安全传输层协议（TLS1.0）承载 HTTP 交换以保证机密性和消息完整性。如果使用了 HTTP POST binding，则在<Response>消息中的<Assertion>元素必定被签字，如果使用的是 HTTP-Artifact binding，则<Assertion>元素可能被签字。

正如本建议书中所说，服务提供商必定要处理<Response>消息和附在<Assertion>上的任何元素。

11.4.1.3.6 服务提供商同意或拒绝用户代理访问

为使协议子集完整，服务提供商处理<Response> 和<Assertion>并且同意或拒绝对资源的访问。服务提供商可能会和用户代理一起使用它所选择的会话机制来设置安全环境。<Assertion>提供的任何接下来的使用都任凭服务提供商和其他的依赖用户处理，以满足它们中使用的任何约束。

11.4.1.4 认证请求协议的使用

这个协议子集基于本建议书中定义的认证请求协议。这里，服务提供商是请求的发起者和依赖用户，而责任人是推荐者、被请求认证的对象和确认实体。还可能有一些额外的依赖用户或者确认实体任凭身份提供者的处理。

11.4.1.4.1 <AuthnRequest>的用法

服务提供商可以包括任何如同在本建议书中描述的消息内容。所有的处理规则也是在建议书中定义的。<Issuer>消息元素必须出现并且包含请求服务提供商的唯一标识符，格式属性必须忽略或者有如下的值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

如果身份提供者不能满足请求，它必须回复一个包含适当的错误状态码的<Response>消息。

如果服务提供商想要允许身份提供者为没有已存许可证的责任人建立一个新的许可证，它必须包含一个将 AllowCreate 属性值设为“true”的<NameIDPolicy>。否则，只有身份提供者先前已经为它建立了服务提供商所使用的许可证的责任人才能够成功的认证。

服务提供商可能在请求中包含一个<Subject>元素，<Subject>元素指定了服务提供商想要收到断言的实际身份认证。这个元素不能包含任何的<SubjectConfirmation>元素。如果身份提供者不能识别这一身份的责任人，它必须回复一个包含错误状态和没有断言的<Response>消息。

<AuthnRequest>消息可能会被签字（如同被所使用的 SAML binding 所掌控）。如果使用了 HTTP Artifact binding，用户的认证是可选的并且 binding 允许的任何机制都可能被使用。

如果<AuthnRequest>请求消息没有被认证或者没有被完整保护，那么它包含的信息就是不可信的 except as advisory。不论请求是不是被签字，身份提供者必须确保请求中的任何<AssertionConsumerServiceURL>或者<AssertionConsumerServiceIndex>元素均被检验是附属于将要接收应答的服务提供商。不这样做可能会导致中间人攻击。

11.4.1.4.2 <Response>的用法

注 1（资料性的）— PE26（见 OASIS PE:2006）提供了为本部分内容提供了详细介绍，详情参见附录八。

如果身份提供者想要返回一个错误信息，它就不能在<Response>应答消息中包含任何断言。否则，如果请求成功了（或如果应答没有和请求相关联），<Response>应答消息元素必须遵循如下：

- <Issuer>元素可能被省略，但是如果出现则必须包含身份提供者发布的唯一标识符；格式属性必须忽略或者有如下的值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

  注 2（资料性的）— PE17 (see OASIS PE:2006)建议把上面的段落替换为：

  如果<Response>消息被签字了或者如果附带的断言被加密了，则<Issuer>元素必须出现。否则它将被忽略。如果出现则必须包含身份提供者发布的唯一标识符；格式属性必须忽略或者有如下的值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。
- 它必须包含至少一个<Assertion>断言。每一个断言的<Issuer>元素必须包含发布此断言的身份提供者的唯一标识符。格式属性必须忽略或者有如下的值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。
- 一个或者多个断言必须包含至少一个可以向身份提供者反映责任人的身份认证的<AuthnStatement>认证断言。
- 至少有一个包含<AuthnStatement>的断言中包括一个<Subject>元素，这个<Subject>元素必须有至少一个含有方法：urn:oasis:names:tc:SAML:2.0:cm:bearer 的<SubjectConfirmation>元素。如第11.4.4节中所定义的，如果身份提供者支持单点注销协议子集，任何这样的认证断言必须包含一个SessionIndex属性来使服务提供商的每一次会话注销请求有效。
- 上面所描述的传递者<SubjectConfirmation>元素必须包含一个<SubjectConfirmationData>元素，这个<SubjectConfirmationData>元素含有一个易接收？的特性，这一特性包括：服务提供商的断言使用者服务（ACS）URL和用来限制断言交付窗口的NotOnOrAfter属性。它可能包含一个限制断言的可交付客户端地址的Address地址属性。它不能包含NotBefore属性。如果包含的消息是在<AuthnRequest>的应答消息中，则InResponseTo属性必须匹配请求ID。
- 其他的断言方法和确认方法可能被包含在身份提供者自行决定的断言之中。特别是，可能包括<AttributeStatement>属性。<AuthnRequest>可能包含一个AttributeConsumingServiceIndex XML特性，此特性涉及第9节中介绍的期望或必须的属性的信息。身份提供者可能会忽略它或者自行决定发送其他的属性。
- 包含一个传递者附属认证的断言必须包含一个<AudienceRestriction>，而这个<AudienceRestriction>包括服务提供商的作为<Audience>的唯一标识符。
- 其他的条件（和其他的<Audience>元素）可能是根据服务提供商的请求或者是由身份提供者自行决定。（当然，所有的条件都必须被服务提供商所理解和接受，才能保证断言是有效的。）即便是要，但身份提供者并不负有责任去遵守<AuthnRequest>中被请求的各种<Conditions>。

11.4.1.4.3 <Response>消息处理规则

注（资料性的）—PE26 (见OASIS PE:2006)为本节提供了详述，详情参见附录八。

不管所使用的 SAML 绑定，服务提供商必须完成下列项：

- 检验出现在断言消息或者应答消息上的任何签字。
- 检验任何承载者<SubjectConfirmationData>元素中的Recipient属性与<Response>或凭证应答被转发到的断言使用者服务（ACS）URL的匹配。
- 检验任何传递者<SubjectConfirmationData>元素中的NotOnOrAfter属性还未传递，以满足提供者之间允许的时钟脉冲相位差。
- 检验传递者<SubjectConfirmationData>中的InResponseTo属性与源<AuthnRequest>消息的ID匹配，除非应答消息是未被请求的，这时此属性不必出现。
- 检验在其他发面的任何依靠的断言都是合法的。
- 如果任何传递者<SubjectConfirmationData>都包含一个Address地址属性，服务提供商可能会检查核实用户代理的客户端地址。

- 任何无效的断言，或者其主体确认需求无法满足的断言应该被丢弃并且不能用来为责任人建立安全关联。
- 如果用来为责任人建立安全关联的<AuthnStatement>包含一个SessionNotOnOrAfter属性，那么一旦到达该时刻，则这个安全关联就应该被丢弃，除非服务提供商通过再次使用该协议子集重建了责任人的身份。

11.4.1.4.4 Artifact-specific <Response>应答消息处理规则

如果使用 HTTP Artifact binding 来转发<Response>应答，对于凭证使用 Artifact 决定协议子集的间接引用必须保证互相证实、完整性保护和机密性。

身份提供者必须保证<Response>应答消息发布给那个服务提供商，此服务提供商才能得到作为<ArtifactResolve>请求结果的消息。

用于间接引用 artifact 的 SAML binding 或者是消息的签字均可被用来鉴别用户和保护消息。

11.4.1.4.5 POST-specific处理规则

注（资料性的）— PE26（见 OASIS PE:2006）为本节内容提供了详述，详情参见附录八。

如果使用 HTTP POST binding 来转发<Response>应答，则附带的断言消息必须被签字。

服务提供商必须保证传递者断言消息不是重复的，通过为时间的长度维持一系列使用过的 ID 数值来实现保证，而因为这个时间，根据<SubjectConfirmationData>中的 NotOnOrAfter 属性，断言会被认为是有效的。

11.4.1.5 未被请求的应答

一个身份提供者可能会通过发送一个未被请求的<Response>应答消息给一个服务提供商来发起这个协议子集。

一个未被请求的<Response>应答不能包含 InResponseTo 属性，任何传递者<SubjectConfirmationData>元素也不能包含此属性。如果使用了元数据，<Response>应答或者其重建消息应该缺省被转发给指定服务提供商的<md:AssertionConsumerService>端点。

特别提及的是，身份提供者可能会包含一个 binding-specific "RelayState" 参数，这个参数表明了在与服务提供商相互一致的基础上，怎样处理随后与用户代理的交互。这可能是服务提供商处一个资源的 URL。服务提供商应该为处理未被请求的应答做些准备，即指明一个缺省的路径来发送用户代理并随后成功处理应答。

11.4.1.6 元数据的使用

本建议书中的第 9 部分定义了一个端点元素<md:SingleSignOnService>，用来描述所支持的绑定和位置，通过这些位置，服务提供商可以发送请求给使用这个协议子集的身份提供者。

<md:IDPSSODescriptor>元素的 WantAuthnRequestsSigned 属性可以被身份提供者用来提供要求被签字的请求。<md:SPSSODescriptor>元素的 AuthnRequestsSigned 属性可以被服务提供商用来提供为它的所有请求签字的意图。

提供者可以提供用来签字的钥匙，使用具有标记使用属性的<md:KeyDescriptor>元素来为请求、应答和断言。当加密 SAML 元素时，具有标记使用属性的<md:KeyDescriptor>元素可以用来提供支持的加密算法和设置，及用来接收重要密钥的公钥。

编入索引的端点元素<md:AssertionConsumerService>用来描述所支持的绑定和位置，通过这些位置, 身份提供者可以发送应答给使用这个协议子集的服务提供商。索引特性用来区分可能的端点，也就是可能会在<AuthnRequest>消息中提及指出的端点。如果没有在请求中指定，那么 isDefault 属性用来指定要使用的端点。

服务提供商使用<md:SPSSODescriptor>元素的 WantAssertionsSigned 属性来提供一个要求在这个协议子集中传送的断言被签字的请求。这是使用特定映射所利用的签字请求的一个补充。身份提供者并不对此负有责任，但是它必须知道一个未签字的断言是不足的。

如果请求或应答消息是使用 HTTP Artifact binding 发送的，那么凭证发布者必须在它的元数据中提供至少一个<md:ArtifactResolutionService>端点元素。

<md:IDPSSODescriptor> 可能包含 <md:NameIDFormat> 、 <md:AttributeProfile> 和 <saml:Attribute>元素，这些元素表现出支持特定名称标识符格式、特性规范或特定属性和值的能力。这种在给定的认证交换中支持任何这样的特性的能力，是依赖于身份提供者的政策和谨慎的。

<md:SPSSODescriptor>元素还可能被用来为将要随同认证信息一起被发送的 SAML 特性提供服务提供商的需求或要求。特性的实际内容通常都由身份提供者自行决定。它的元数据中可能包含一个或更多 <md:AttributeConsumingService>元素，每个元素都有一个索引属性来区分可能在<AuthnRequest>消息中提及指定的不同的服务。isDefault 属性用来指定一些缺省的特性请求。

11.4.2 增强型客户端或代理（ECP）协议子集

一个*增强型客户端或代理*（ECP）是一个知道如何联系恰当的身份提供者的系统实体，可能以依赖情景的方式，并且还支持反转的 SOAP（PAOS）binding（见第 10 节）。

本协议子集所使能的情景举例如下：一个责任人使用 ECP 规范去访问服务提供商处的资源，或者访问身份提供者，这样服务提供商和被请求的资源能够理解请求。责任人经身份提供者认证（或者已经被认证），身份提供者随即发送一个认证断言（可能伴随着从服务提供商发来的输入）。然后服务提供商使用断言并且随即为责任人建立一个安全关联。在这个过程中，一个名称标识符也可能会为责任人建立在提供者之间，依于交互作用的参数和责任人的同意。

这个协议子集是基于与 PAOS binding 联合在一起的 SAML 认证请求协议的。

注 — 责任人经由身份提供者认证的方法在SAML讨论的范畴之外。

11.4.2.1 必需的信息

证明：urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp（这也是附件 A 中相应的 ECP 规范所指定的目标名称段。）

联系方式：security-services-comment@lists.oasis-open.org

SAML 证实方法标识符：SAML V2.0 "承载者" 证实方法标识符 urn:oasis:names:tc:SAML:2.0:cm:bearer 由本协议子集使用。

说明：见下文。

更新：无。

11.4.2.2 协议子集概述

如上所述，ECP 协议子集指定了在增强型客户端或代理和服务提供商及身份提供者之间的交互活动。它是第 11.4.1 节中描述的 SSO 协议子集的特殊应用。否则，如果没有被协议子集所指定，并且如果没有特别针对基于浏览器的绑定的使用，则第 11.4.1 节中所指定的规则必须遵守。

ECP 是满足以下两个条件的客户端或者代理：

- 它具有，或者知道怎么去得到，在与服务提供商的交互活动的环境中，关于与责任人相联系的且ECP想要使用的身份提供者的信息。

这将允许服务提供商产生认证请求给 ECP，而不必去知道和发现恰当的身份提供者（快速有效地到达了第 11.4.1 节中 SSO 协议子集的第 2 步）。

- 可以使用反转SOAP（PAOS）绑定作为认证请求和应答的协议子集。

这使得服务提供商可以通过 ECP 获得认证断言，此 ECP 即不是直接交互，否则（也就是直接交互活动的情景之外）必须是直接可设定地址的，也不是持续可使用的。当使用已定义好的交换形式和协议子集去实现互用性时，它还起到协调 SOAP 的利益的杠杆作用。ECP 可以看做是一个在服务提供商和身份提供者之间的 SOAP 仲裁者。

一个*增强型客户端*可能是支持本协议子集中描述的功能的浏览器或者某些其他的用户代理。一个*增强型代理*是一个效仿增强型客户端的 HTTP 代理。除非做出说明，否则，所有提及增强型客户端的陈述都将被理解为即是关于增强型客户端的也是关于增强型客户端代理的陈述。

由于增强型客户端通过 HTTP 请求和应答载体来发送和接收消息，所以它对协议消息的大小没有主观约束力。

此协议子集协调反转 SOAP（PAOS）绑定（见第 10 节）。此协议子集的贯彻执行必须遵守在此映射中指明 PAOS 所支持的 HTTP 迹象的规则，作为在此协议子集中所指定的规则的补充。此协议子集利用一个 PAOS SOAP 标题头块在 HTTP 响应端和 ECP 之间传递，但是却不定义 PAOS 本身。此协议子集定义了伴随着 SAML 请求和应答的 SOAP 标题头块。这些标题头块可能是由其他必需的 SOAP 标题头块组成的，比如在需要增加安全性时可由 SOAP 消息安全标题头块组成，再比如用于认证请求的数字签字。

有两类请求/应答 SOAP 标题头块在使用：对应一般的 PAOS 信息的 PAOS 标题头块，和为 ECP 规范的功能性传递特定信息的 ECP 协议子集特殊标题头块。

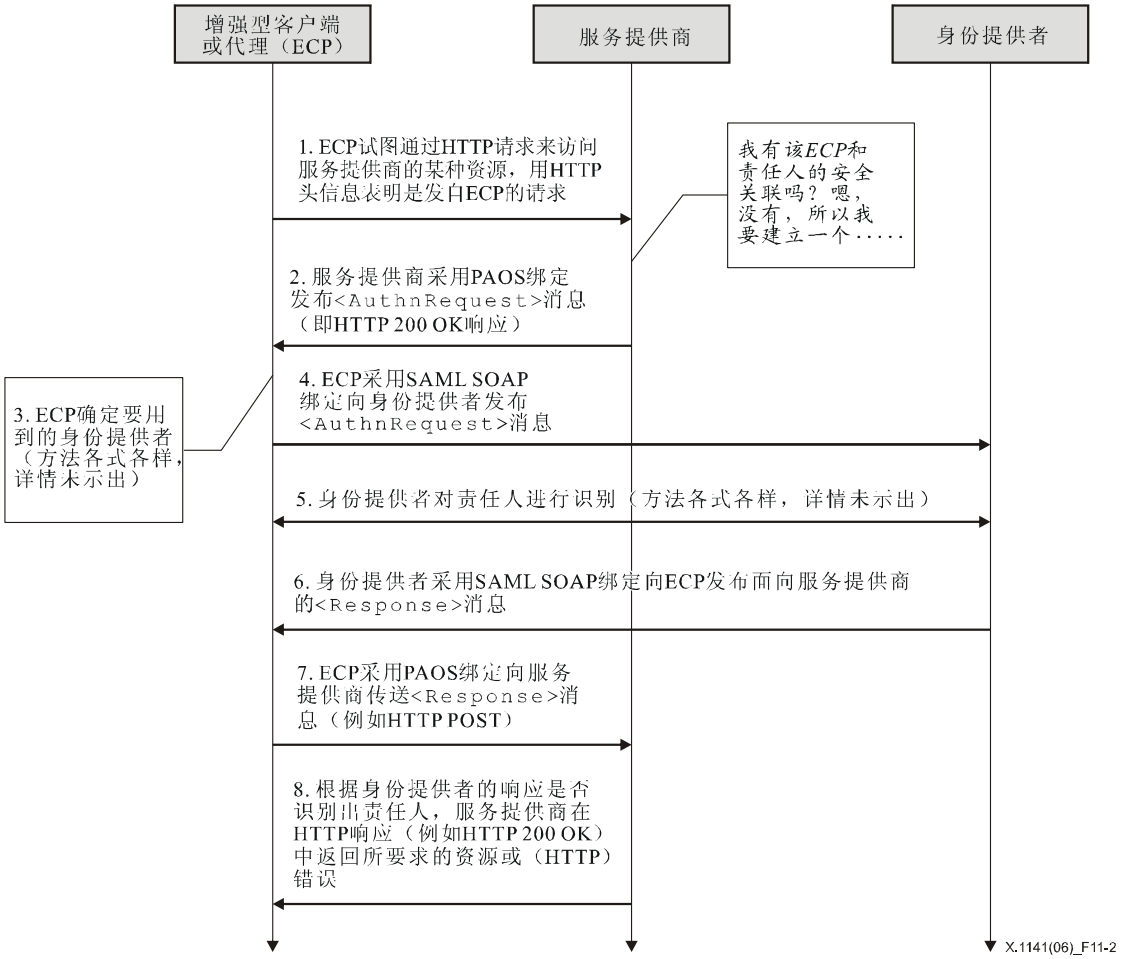

图 11-2/X.1141－ECP协议子集的处理流程

图 11-2 图示阐明了 SSO 使用 ECP 的基本模版。接下来的步骤经由规范所描述。在一个单独的步骤中，可能会有一个或者更多基于本步骤中所使用映射的实际消息交换和其他的从属执行的行为。

1） ECP发布HTTP请求给服务提供商

步骤1中，责任人通过ECP制作发送一个HTTP请求给服务提供商处的一个安全资源，而服务提供商处还没有为ECP和责任人建立的安全关联。

2） 服务提供商发布<AuthnRequest>认证请求给ECP

步骤2中，服务提供商发布一个<AuthnRequest>消息给ECP，而ECP将此消息转发给适当的身份提供者。此处使用了反转SOAP（PAOS）绑定（见第10节）。

3） ECP确定身份提供者

步骤3中，ECP获得一个身份提供者端点的位置，此身份提供者需得适合于支持ECP的首选映射的认证请求协议。完成这个任务的方法是依赖执行。ECP会使用第11.4.3节中描述的SAML身份提供者发现协议子集。

注资料性的 — PE18（见 OASIS PE:2006）建议从上面段落中删掉最后一行。

4） ECP传递<AuthnRequest>认证请求给身份提供者

步骤4中，ECP传递<AuthnRequest>认证请求给第3步中确定的身份提供者，这会利用到SAML SOAP 绑定（见第10节）的改进形式，额外，身份提供者可能会在答复SAML请求之前与ECP交换一些独有的HTTP消息。

5） 身份提供者验证责任人

步骤5中，身份提供者使用本规范范畴以外一些方法对责任人进行身份认证。这可能需要一个新的认证行为或者重用一个已有的认证方法。

6） 身份提供者发布<Response>应答给ECP，转发目的地址是服务提供商

步骤6中，身份提供者使用SAML SOAP 绑定发布一个<Response>消息，此消息将经由ECP转发给服务提供商。此消息可能指明一个认证错误或者包含（至少）一个认证断言。

7） ECP传递<Response>消息给服务提供商

步骤7中，ECP使用PAOS映射传递<Response>消息给服务提供商。

8） 服务提供商同意或拒绝责任人的访问

步骤8中，从身份提供者处收到<Response>应答消息后，服务提供商或者是为责任人建立一个自己的安全关联并返回请求的资源，或者是回复给责任人ECP一个认证错误。

11.4.2.3 协议子集描述

下面的段落分别对单独的步骤进行了详述。

11.4.2.3.1 ECP发布HTTP请求给服务提供商

ECP 发送一个 HTTP 请求给服务提供商，指定一些资源。这个 HTTP 请求必须符合 PAOS 映射，即它必须包含下面的 HTTP 标题头字段：

1） HTTP Accept标题头域标志着接收MIME类型的能力“application/vnd.paos+xml”

2） HTTP PAOS标题头域最小化地指明了PAOS版本：urn:liberty:paos:2003-08

3） 此外，支持本协议子集必须在HTTP PAOS标题头域中以服务属性值的形式指明，此属性的值为：urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp。这个属性值必须符合PAOS Request SOAP标题头块中的服务特性。

例如，用户代理从服务提供商处请求一个页面的方式如下：

```
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'
```

11.4.2.3.2 服务提供商发布<AuthnRequest>认证请求给ECP

当服务提供商在允许责任人访问指定的资源前为它请求安全关联时，也就是，在提供服务和数据之前，它可以使用 PAOS 绑定回复一个包含<AuthnRequest>消息的 HTTP 应答给 HTTP 请求。服务提供商将发布一个包含单独的 SOAP 封装的 HTTP 200 OK 应答给 ECP。

这个 SOAP 封装必须包括：

1） 一个在SOAP消息体中的<AuthnRequest>元素，给最终的SOAP接收端-身份提供者-使用。

2） 一个目标地址是ECP的PAOS SOAP标题头块，它所使用的SOAP参与者属性值为：http://schemas.xmlsoap.org/soap/actor/next。这个标题头块提供了控制信息，比如在这个恳求-应答消息交换模式中应答要被发送的URL地址。

3） 一个目标地址是ECP的ECP 协议子集特殊Request SOAP标题头块，它所使用的SOAP参与者属性值为：http://schemas.xmlsoap.org/soap/actor/next。ECP Request标题头定义了与ECP可能将要处理的认证请求联系在一起的信息，比如服务提供商可接受的一组身份提供者，ECP是否会通过客户端与责任人交互，以及可能会被呈现在责任人面前并可被理解的服务提供商的名称。

这个 SOAP 封套也可能会包含一个目标地址是 ECP 的 ECP RelayState SOAP 标题头块，它所使用的 SOAP 参与者属性值为：http://schemas.xmlsoap.org/soap/actor/next。这个标题头包含了将会与 SAML 应答一起被 ECP 送回的信息。

11.4.2.3.3 ECP确定身份提供者

ECP 将决定哪个身份提供者是合适的并适当地将 SOAP 消息传送。

11.4.2.3.4 ECP发布<AuthnRequest>请求给身份提供者

在将<AuthnRequest>消息传递给身份提供者之前，ECP 必须先将 PAOS、ECP RelayState 和 ECP Request 标题头块移除，使用 SAML SOAP 绑定的改进形式。SAML 请求通过一般样式的 SOAP 传递，但是身份提供者对 ECP 的 HTTP 请求回复一个 HTTP 应答，此应答可能会包含，比如，一个 HTML 注册表单或者一些其他的针对介绍的应答。可能会发生一系列的 HTTP 交换，但是最后身份提供者必须完成 SAML SOAP 交换并且通过 SOAP 绑定返回一个 SAML 应答。

<AuthnRequest>元素本身可能会被服务提供商所签字。在这些及其他方面，必须依照在第 11.4.1.4.1 节中浏览器单点登录 SSO 规范所指定的消息格式。在本步之前或随后，身份提供者必须通过一些方法建立了责任人的身份认证，否则它必须返回一个错误<Response>应答，如同后面第 11.4.2.3.6 节中描述的那样。

11.4.2.3.5 身份提供者验证责任人

在前一步中或者随后的某个时候，身份提供者必须建立了责任人的身份认证（除非它返回了一个错误消息给服务提供商）。强制请求<AuthnRequest>特性的值如果是 true，则身份提供者有责任从新建立这个身份认证，而不能仍依靠责任人可能已具有的身份状态。另外在其他方面，身份提供者可以使用任何方法去认证用户代理，依照以<RequestedAuthnContext>元素的形式出现在<AuthnRequest>请求中的任何要求。

11.4.2.3.6 身份提供者发布<Response>应答给ECP，转发目的地址为服务提供商

身份提供者建立了责任人的身份认证之后，返回一个针对认证请求的 SAML<Response>应答消息（或者 SOAP 错误）。SAML 应答的传送基于在 SOAP 消息中使用 SAML SOAP 绑定，这个 SOAP 消息的消息体中包含一个<Response>元素，传送给作为最终 SOAP 接收端的服务提供商使用。必须遵守第 11.4.1.4.2 节中浏览器单点登录 SSO 规范所指定的应答规则。

身份提供者的应答消息必须包含协议子集特殊 ECP Response SOAP 标题头块，并且可能包含一个 ECP RelayState 标题头块，它们的目的地址都是 ECP。

11.4.2.3.7 ECP传送<Response>消息给服务提供商

ECP 将标题头块移除，并且可能添加一个 PAOS Response SOAP 标题头块和 ECP RelayState 标题头块，然后利用 PAOS 绑定将 SOAP 应答传给服务提供商。

在给服务提供商的应答中，<paos:Response> SOAP 标题头块一般用来将这个应答与之前从服务提供商处来的请求联系起来。在这个协议子集中，相关性 refToMessageID 属性并不是必须的，因为 SAML <Response>元素的 InResponseTo 属性可以用来达成此目的，但是如果<paos:Request> SOAP 标题头块含有一个 messageID，则<paos:Response> SOAP 标题头块就必须使用。

<ecp:RelayState>标题头块值与服务提供商的请求一起是由服务提供商特别提供给 ECP 的，但是如果身份提供者产生的是一个未被请求的应答（没有收到相应的 SAML 请求），则它可能会包含一个 RelayState 标题头块，此标题头块基于与服务提供商达成的相互约定来指出怎样处理随后与 ECP 的交互。这可能就是服务提供商处资源的 URL。

如果服务提供商在给 ECP 的请求中包含一个<ecp:RelayState> SOAP 标题头块，或者如果身份提供者在它的应答中包含一个<ecp:RelayState> SOAP 标题头块，则 ECP 必须在给服务提供商的 SAML 应答中包含一个同样的标题头块。服务提供商赋予这个标题头块（如果有）的值必须是优先考虑的。

11.4.2.3.8 服务提供商允许或拒绝责任人的访问

一旦服务提供商收到了在 HTTP 请求（在使用 PAOS 的 SOAP 封套中）中的 SAML 应答，它可能会在 HTTP 应答中回复服务数据。在应答之中，必须遵守第 11.4.1.4.3 节和第 11.4.1.4.5 节中的浏览器 SSO 单点登录规范所指定的规则。也就是，利用 HTTP POST 绑定接收应答<Response>时与利用 PAOS 使用同样的处理规则。

11.4.2.4 ECP协议子集Schema的使用方法

ECP 协议子集 XMLSchema 定义了此协议子集使用的 SOAP Request/Response 请求/应答标题头块。下面是此 Schema 文档的完整清单。

```
<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 identity: 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>
```

下面的部分描述了如何使用这些 XML 结构

11.4.2.4.1 PAOS 请求标题头块: SP 到 ECP

PAOS 请求标题头块发信号通知使用 PAOS 处理并包含了下面的属性：

— `responseConsumerURL` [必需的]

指明 ECP 要将错误应答发送的地址。也用于核实身份提供者应答的正确性，方法是将这个地址与 ECP 应答标题头块中的 `AssertionServiceConsumerURL` 对比交叉检查。这个属性值必须跟在 `<AuthnRequest>`请求中传送的 `AssertionServiceConsumerURL`(或元数据中引用的 URL)一致。

注（资料性的）— PE22（见OASIS PE:2006）建议改最后一行的`AssertionServiceConsumerURL`为`AssertionConsumerServiceURL`。

— `service` [必需的]

指明将要使用的 PAOS 服务是 SAML 认证规范。属性值必须是 `urn:oasis:names:tc:SAML:2.0:profiles:SSO:ecp`。

— `SOAP-ENV:mustUnderstand` [必需的]

属性值必须为 1（true）。如果 PAOS 标题头块没有被理解则产生一个 SOAP 错误。

— `SOAP-ENV:actor` [必需的]

属性值必须为 `http://schemas.xmlsoap.org/soap/actor/next`。

— `messageID` [可选的]

允许可选任意应答的相关性。它可用在此协议子集中但不是必须的，因为这项功能由 SAML 协议层提供，通过`<AuthnRequest>`请求中的 ID 属性和`<Response>`应答中的 `InResponseTo` 属性。

PAOS Request SOAP 标题头没有元素内容

11.4.2.4.2 ECP 请求标题头块: SP 到 ECP

ECP Request SOAP 标题头块用来传送 ECP 处理认证请求所需要的信息。它是强制性的并且它的出现标志着使用了这个协议子集。它包括以下元素和属性：

— `SOAP-ENV:mustUnderstand` [必需的]

此属性值为 1（true）。如果 ECP 标题头块没被理解则会产生一个 SOAP 错误。

— `SOAP-ENV:actor` [必需的]

属性值为 `http://schemas.xmlsoap.org/soap/actor/next`。

— `ProviderName` [可选的]

这是一个为发出请求的服务提供商准备的人类易懂名称。

— `IsPassive` [可选的]

这是一个布尔类型的值。如果是 true，则身份提供者和客户端自身不能控制来自于请求发起者的用户界面，并且不能以一种明显的形式与责任人进行交互。如果没有为此属性提供值，则缺省为 true。

— `<saml:Issuer>` [必需的]

这个元素必须包含所请求服务的提供者的唯一标识符；格式特性会被忽略或者含有数值 `urn:oasis:names:tc:SAML:2.0:nameid-format:entity`。

— `<samlp:IDPList>` [可选的]

身份认证的选择列表提供了服务提供商要查看的信息并且 ECP 会从中选择针对请求的处理信息。

11.4.2.4.3 ECP RelayState 标题头块: SP到 ECP

ECP RelayState SOAP 标题头块用来传送从服务提供商来的状态信息，稍后处理从 ECP 来的应答时会需要此状态信息。这个是可选的，但是如果使用了的话，ECP 就必须包含一个在第 5 步的应答中一样的标题头块。它含有以下特性：

注（资料性的）— PE27（见OASIS PE:2006）建议在上面文本中以第7步代替第5步。

— `SOAP-ENV:mustUnderstand [Required]`

属性值需为 1（true）。如果标题头块没有被理解则需产生一个 SOAP 错误。

— `SOAP-ENV:actor` [必需的]

属性值需为 `http://schemas.xmlsoap.org/soap/actor/next`。

标题头块元素的内容是请求端创建的包含状态信息的字符串。如果提供了这一属性，则 ECP 在第 5 步应答服务提供商时必须包含与 RelayState 标题头块同样的属性值。字符串属性值的长度不能超过 80 字节，并且请求端需为其提供完整性保护，此完整性保护独立于任何其他的在消息传递中可能存在或不存在的保护。

下面是一个示例，从服务提供商到 ECP 的 SOAP 认证请求：

```
<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"
SOAP-ENV:actor="http://schemas.xmlsoap.org/soap/actor/next"
SOAP-ENV: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>
```

如上所述，在认证请求被传递给身份提供者之前，ECP 将 PAOS 和 ECP 标题头块从 SOAP 消息中移除了。下面是一个从 ECP 到身份提供者的认证请求的例子：

```
<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>
```

11.4.2.4.4 ECP Response 标题头块: IdP 到 ECP

ECP 应答 SOAP 标题头块必须被用在从身份提供者到 ECP 的应答上。它包含如下特性：

— `SOAP-ENV:mustUnderstand` [必备的]

此属性必须为 1（true）。如果 ECP 标题头块没有被理解则将产生一个 SOAP 错误。

— `SOAP-ENV:actor` [必备的]

此属性必须为 `http://schemas.xmlsoap.org/soap/actor/next`。

— `AssertionConsumerServiceURL` [必备的]

此属性由身份提供者设置，此设置基于`<AuthnRequest>`认证请求消息或者身份提供者将获得的服务提供商的元数据。

ECP 必须确认此属性值与它从 `responseConsumerURL` 中获得的属性值保持一致，而这个 `responseConsumerURL` 是 ECP 从服务提供商处收到的 PAOS Request SOAP 标题头块中的。由于 `responseConsumerURL` 可能是相对的，而 `AssertionConsumerServiceURL` 是绝对的，所以可能需要一些处理/标准化。

这一机制用于一种安全用途，用来确认正确的应答目的地。如果属性值不匹配，则 ECP 将产生一个 SOAP 错误应答给服务提供商并且不返回 SAML 应答。

ECP Response SOAP 标题头没有元素内容。

下面是一个 IdP-to-ECP 应答的例子：

```
<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_assert
ion_consumer"/>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <samlp:Response> ... </samlp:Response>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

11.4.2.4.5 PAOS Response 标题头块: ECP到SP

PAOS Response 标题头块包含如下特性：

— `SOAP-ENV:mustUnderstand` [必备的]

属性值必须为 1（true）。如果 PAOS 标题头块没有被理解则产生一个 SOAP 错误。

— `SOAP-ENV:actor` [必备的]

属性值为 `http://schemas.xmlsoap.org/soap/actor/next`。

— refToMessageID [可选的]

允许与 PAOS 请求的相关性。如果相应的 PAOS 请求指定了 messageID 特性，则这个可选属性（跟标题头块是个整体）必须被 ECP 添加。在 SAML 中使用<AuthnRequest>和<Response>相关性来提供同样的功能。

PAOS Response SOAP 标题头没有元素内容。

下面是一个 ECP-to-SP 应答的例子：

```
<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>
```

11.4.2.5 安全性考虑

<AuthnRequest>认证请求消息应该被签字。浏览器 SSO 单点登录规范所指定的每一个规则，<Response>应答消息中所附带的断言都必须被签字。在 SOAP 封装中通过 PAOS 对应答消息的传递与使用 HTTP POST 绑定在本质上是类似的，并且对所使用映射的安全对策是恰当的。

SOAP 标题头必须具备完整性保护，比如包含 SOAP 消息安全或者通过在与客户端的每一次 HTTP 交换中使用 TLS（安全传输层协议）。

对于 ECP，服务提供商应该被鉴别，比如使用服务器端 TLS 认证。

对于身份提供者，ECP 应该被鉴别，比如通过维持一个认证会话。在身份提供者返回一个<Response>应答消息之前，传送<AuthnRequest>请求消息之后的任何 HTTP 交换都必须与源请求建立安全关联。

注（资料性的）— PE20（见OASIS PE:2006）建议增加一段对ECP Metadata Considerations的讨论，如下。

最好是把第11节中浏览器单点登录SSO规范所指定的规则也放在这里。特别是，被编入索引且带有一个urn:oasis:namees:tc:SAML:2.0:bindings:PAOS映射的端点元素<md:AssertionConsumerService>，可能会被用于描述所支持的映射及身份提供者使用此规范给服务提供商发送应答消息的路径。并且，带有urn:oasis:namees:tc:SAML:2.0:bindings:SOAP映射的端点元素<md:SingleSignOnService>，可能会被用于描述所支持的映射以及服务提供商使用此规范给身份提供者发送请求消息的路径。

11.4.3 身份提供者发现规范

本段定义了一个规范，服务提供商使用此规范可以发现责任人在其网络浏览器 SSO 规范中使用的是哪一个身份提供者。由于部署中有很多身份提供者，服务提供商需要一种方法来发现责任人使用的是哪一个身份提供者。此发现规范基于部署中的 cookie，而 cookie 是要写到一个输入域中且此输入域在身份提供者和服务提供商之间是公共的。配置所预定的输入域在规范中被称为公共域，并且包含身份提供者列表的 cookie 被称为公共域 cookie。

公共域中的哪一个实体主机网络服务器是部署的发起者不在本规范所讨论的范畴。

注（资料性的）— PE32（见OASIS PE:2006）建议增加如下内容来描述必需的信息：

标识：urn:oasis:names:tc:SAML:2.0:profiles:SSO:idp-discovery

联系方式：security-services-comment@lists.oasis-open.org

11.4.3.1 公共域cookie

Cookie 的名字为"_saml_idp"。Cookie 属性值的格式是一组单个或者多个基 64 编码的 URL 值，这些值之间由单独空格字符分开。每个 URI 都是一个身份提供者的唯一标识符，如同第 7 节中定义的那样。最后的一组属性值就是 URL 编码。

公共域 cookie 写入服务应该把身份提供者的唯一标识符添加到列表中。如果标识符已经存在于列表中了，它将被移除然后重新添加。目的是确保最近建立的身份提供者会话是列表中最新的一个。

cookie 应该被设成以路径前缀"/"开始。输入域必须被设成".[common-domain]"，此处的[common-domain]是指为使用这个规范而建立在 deployment 内的公共域。必须有一个优先的周期。cookie 必须被安全标记。

cookie 的句法应该与 IETF RFC 2965 保持一致。cookie 可以是随会话改变或者持久不变。这个选项可以在 deployment 中设定，但是必须统一使用于 deployment 中的所有身份提供者。

11.4.3.2 设置公共域cookie

身份提供者认证了一个责任人之后，它可能会设置公共域 cookie。身份提供者设置 cookie 的方法是特定实施的，只要根据上面提供的参数成功设置 cookie 即可。下面是一个可行的执行策略但不是标准的：

- 身份提供者可能已经在公共域中为自己建立了DNS和IP别名。
- 利用DNS别名重定向用户代理到它自身，使用URL标志"https"作为URL的配置。对于执行来讲URL的结构是私有的并且可能包含认证用户代理所需的会话信息。
- 使用上面指定的参数为重定向用户代理设置cookie。
- 重定向用户代理到它自身，或者如果合适的话，到服务提供商。

11.4.3.3 获得公共域cookie

当服务提供商要去发现责任人使用的是哪个身份提供者时，它调用了一种交换，而此设计交换是用来将公共域中被 HTTP 服务器读取过的公共域 cookie 呈现给服务提供商的。

如果在公共域中的 HTTP 服务器是由服务提供商来运作的或者如果尚有其他的安排，服务提供商可以利用公共域中的 HTTP 服务器去传递它的`<AuthnRequest>`认证请求给身份提供者以获得一个最优化的单点登录 SSO 处理。

服务提供商读取 cookie 的特定方法就是特定实施，只要它能够促使用户代理将按照第 11.4.3.1 节中所提供参数设置的 cookie 呈现即可。下面是一个可行的执行策略但不是标准的：

- 已经在公共域中为自己建立了DNS和IP别名。
- 利用DNS别名重定向用户代理到它自身，使用URL标志"https"作为URL的配置。对于执行来讲URL的结构是私有的并且可能包含认证用户代理所需的会话信息。
- 重定向用户代理到它自身，或者如果合适的话，到身份提供者。

11.4.4 单点注销规范

一旦责任人通过了身份提供者的认证，认证实体可能会与责任人建立一个会话（主要的方法是 cookie、URL 重写或者其他的一些特定执行方法）。基于这个认证事件，身份提供者可能随后发布断言给服务提供商或者其他的依赖用户；依赖用户可以使用此断言与责任人建立自己的会话。

在这个情况下，身份提供者是会话管理者而依赖用户是会话参与者。稍后，在会话管理者所管理的给定会话中，责任人可能会想要终止它与其中的一个单独会话参与者的会话，或者是与所有会话参与者的会话。从前的用例不在本建议书的研究范畴之内。但是，后面的用例适合使用 SAML 单点注销协议规范（见第 11.4 节）。

责任人（或者管理员终止了责任人的会话）可以选择通过联系会话管理者来终止这个"global"会话，或者是单独的会话参与者。在依赖用户与另外一个身份提供者交换关于此责任人的断言的情况下，作为会话管理者的身份提供者自身也可以是会话参与者。

本规范允许协议与同步映射结合在一起，比如 SOAP 映射，或者与异步"front-channel"映射结合在一起，比如 HTTP Redirect、POST 或 Artifact bindings。front-channel 映射可能是必需的，比如，责任人的会话状态单独的以 cookie 的形式存在于用户代理中的情况，并且需要一个在用户代理和会话参与者或者会话管理者之间的直接交互。正如下面所讨论的，在开始协议子集时，如果可以的话，会话参与者需要使用"front-channel"映射来增加会话管理者传播注销成功给所有参与者的可能性。

11.4.4.1 必需的信息

标识：`urn:oasis:names:tc:SAML:2.0:profiles:SSO:logout`

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

11.4.4.2 协议子集概述

图 11-3 图示阐明了实现单点注销的基本模版：

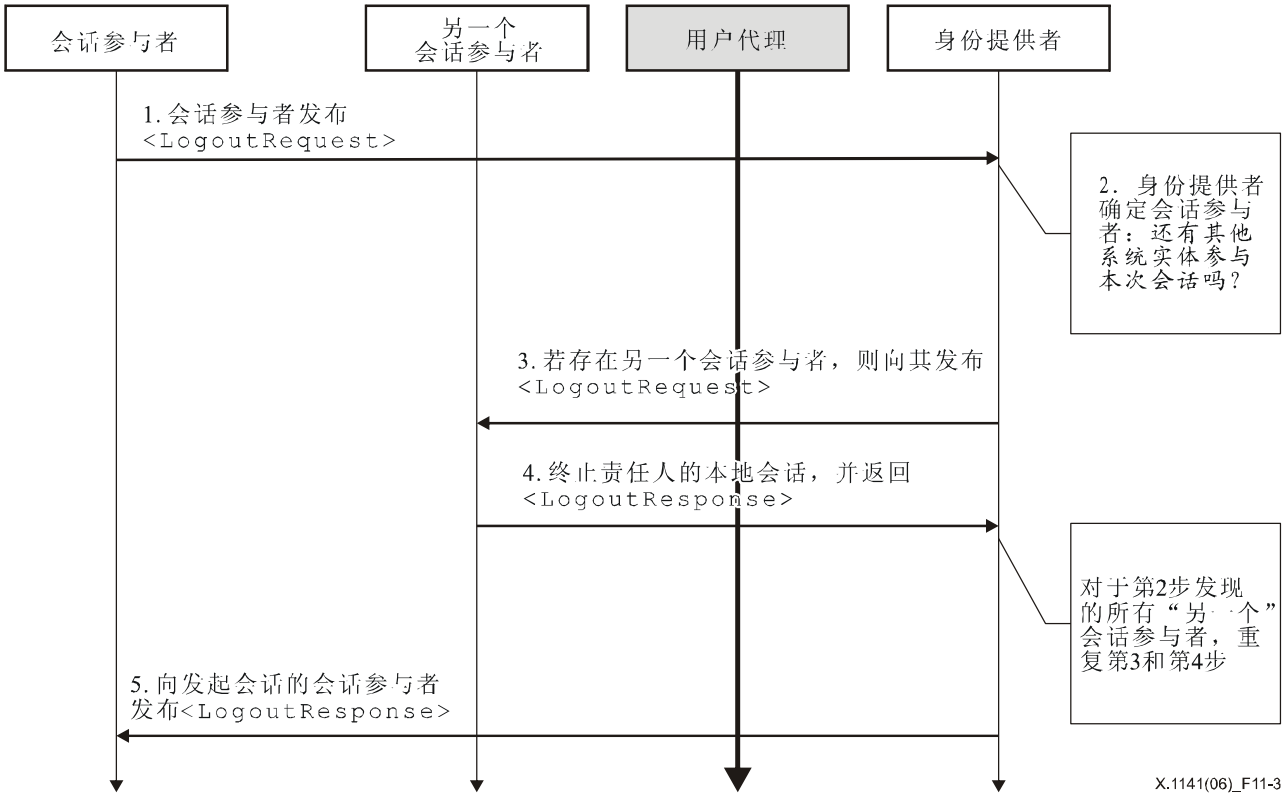

图 11-3/X.1141－实现单点注销的模版

依靠用来贯彻协议子集的 SAML 绑定，已注销用户代理阐明了消息交换可以通过用户代理或者在系统实体之间直接交换。

协议子集描述了以下步骤。在每一个单独的步骤中，可能会有一个或者更多基于本步骤中所使用映射的实际消息交换和其他的从属执行的行为：

1） 会话参与者发布<LogoutRequest>注销请求给身份提供者

第1步中，通过发送<LogoutRequest>注销请求消息给提供相应认证断言的身份提供者，会话参与者发起了单点注销并且终止了责任人的会话。这个注销请求消息可以直接发送给身份提供者也可以通过用户代理间接的发送。

2） 身份提供者确定会话参与者

第2步中，身份提供者使用<LogoutRequest>消息（或者是如果开始了自身的注销，其他的一些机制）中的内容来确定被终止的会话。如果不再有其他的会话参与者，协议子集依第5步处理。否则重复第3步和第4步来识别会话参与者。

3） 身份提供者发布<LogoutRequest>消息给会话参与者/管理者

第3步中，身份提供者发布<LogoutRequest>消息给与一个或多个要终止的会话有关系的会话参与者或会话管理者。请求消息可以直接发送给实体或者是通过用户代理间接发送（如果与第1步中的请求表单保持一致）。

4） 会话参与者/管理者发布<LogoutResponse>应答消息给身份提供者

第4步中，会话参与者或会话管理者根据请求终止了责任人的会话（如果可能）并且返回一个<LogoutResponse>应答消息给身份提供者。应答消息可以直接返回给身份提供者或者是通过用户代理间接返回（如果与步骤三中的请求表单保持一致）。

5） 身份提供者发布<LogoutResponse>应答消息给会话参与者

第5步中，身份提供者发布<LogoutResponse>应答消息给最初发起注销请求的会话参与者。应答消息可以直接返回给会话参与者或者是通过用户代理间接返回（如果与第1步中的请求表单保持一致）。

身份提供者（作为会话管理者）可以在第2步发起这个协议子集并且发布<LogoutRequest>注销请求给所有的会话参与者，同样跳至第5步处理。

11.4.4.3 协议子集描述

如果协议子集是由会话参与者发起的，从第 11.4.4.3.1 节开始。如果是由身份提供者发起的，从第 11.4.4.3.2 节开始。如下所述：

— **单点注销服务**

这是单点注销协议在身份提供者或会话参与者处的端点，这也是<LogoutRequest>注销请求消息或<LogoutResponse>注销应答消息（或者是它们的重建消息）要交付的地方。为请求和应答使用了相同或不同的端点。

11.4.4.3.1 会话参与者发布<LogoutRequest>注销请求给身份提供者

如果注销协议子集是由会话参与者发起的，它（会话参与者）检查所收到的认证断言，这些断言是依附于将要终止的会话的，它还汇总从身份提供者收到的 SessionIndex 属性值。如果与多个身份提供者有关，这协议子集将为每一个单独重复。

为了发起这个协议子集，会话参与者发布<LogoutRequest>注销请求消息给身份提供者的单点注销服务请求端点，此消息包含一个或多个适用的<SessionIndex>元素。至少要包含一个元素。元数据可用于确定此端点的位置和身份提供者支持的映射。

异步映射（Front-Channel）

会话参与者需要使用异步映射，比如 HTTP Redirect,、POST 或者 Artifact 映射（见第 10 节），通过用户代理发送请求给身份提供者（如果责任人的用户代理出现）。接下来身份提供者需要把所有必需的注销消息传播给另外涉及的会话参与者，使用同步或异步映射均可。对于源请求来讲使用异步映射是首选，因为它可以使身份提供者在第 3 步中以最大的成功概率将注销消息传播给其他的会话参与者。

如果使用了 HTTP Redirect 或 POST binding，那么<LogoutRequest>消息在本步中将被交付给身份提供者。如果使用了 HTTP Artifact binding，身份提供者将使用在第 11.4.6 节中定义的凭证解析协议子集，这将会利用比如 SOAP 映射产生了一个给会话参与者的回叫消息以求重新获得<LogoutRequest>消息。

推荐将本步中的 HTTP 交换应用在安全传输层协议 TLS 1.0 之上以求保持机密性和消息的完整性。如果使用了 HTTP POST 或 Redirect 映射，则<LogoutRequest>注销请求消息需要被签字。如果使用了 HTTP Artifact 绑定，当重建被废弃时，还会提供一个鉴别请求发起者的替代方法。

每一种映射都提供了 RelayState 机制，会话参与者可以使用这种机制来建立协议子集交换与源请求的联系。会话参与者应该在 RelayState 属性值中暴露尽可能少的信息，除非使用的协议子集没有机密尺度的要求。

同步映射（Back-channel）

另一种选择，会话参与者可以使用同步映射来直接发送请求消息给身份提供者，比如 SOAP binding（见第 10 节）。然后身份提供者需要利用同步映射将所有必须的注销消息传播给另外涉及的会话参与者。请求端必须已经通过了身份提供者的认证，通过给<LogoutRequest>消息签字实现或者利用所支持的其他映射机制实现。

第 11.4.4.4.1 节中有关于<LogoutRequest>请求消息内容的协议子集特殊规则。

11.4.4.3.2 身份提供者确定会话参与者

如果注销协议子集是由身份提供者发起的，或者身份提供者收到一个有效的<LogoutRequest>消息后处理这个请求，身份提供者必须检查标识符和<SessionIndex>元素并且确定要终止的那些会话。

然后身份提供者针对参与到终止会话中的每一个实体执行第 3 步和第 4 步，除了（发出）源（注销）请求的会话参与者（如果有），就如同第 8.2.7 节中描述的那样。

11.4.4.3.3 身份提供者发布<LogoutRequest>注销请求消息给会话参与者/管理者

为了传播注销，身份提供者发布它自己的<LogoutRequest>请求消息给一个要被终止的会话所涉及的会话管理者或者参与者。这个请求利用 SAML 绑定传送，以求与响应端的性能和身份提供者处用户代理的有效性相一致。

一般来讲，第 1 步中收到源注销请求的映射并没有指明在这个步骤中会使用到的映射，除非在步骤 1 中有标注，利用一个同步映射绕过用户代理迫使身份提供者使用一个同样的映射去传播额外的请求。

第 11.4.4.4.1 节中包含了针对<LogoutRequest>消息内容协议子集的特定规则。

11.4.4.3.4 会话参与者/管理者发布<LogoutResponse>应答消息给身份提供者

如同第 8.2.7 节中定义的那样，会话参与者/管理者必须处理<LogoutRequest>注销请求消息。处理过消息或者遇到错误之后，实体必须再发布一个包含适当状态码的<LogoutResponse>注销应答消息给发出请求的身份提供者，使身份提供者完成 SAML 协议交换。

同步映射（Back-channel）

如果身份提供者使用了同步映射，比如 SOAP 映射（见第 10 节），则应答消息被直接返回以完成同步通信。响应端必须已经通过了发出注销请求的身份提供者的身份认证，通过给<LogoutRequest>消息签字实现或者利用所支持的其他映射机制实现。

异步映射（Front-channel）

如果身份提供者使用了异步映射，比如 HTTP Redirect、POST 或 Artifact 映射（见第 10 节），则<LogoutResponse>应答消息（或者其重建）通过用户代理返回给身份提供者的单点注销服务应答端点。使用元数据来确定此端点的位置和身份提供者所支持的映射。被双方实体均支持的任何异步映射都可能会被使用。

如果使用了 HTTP Redirect 或 POST 映射，则<LogoutResponse>应答消息在本步中被交付给身份提供者。如果使用了 HTTP Artifact 绑定，身份提供者将使用第 11.4.6 节中定义的重建决定规范（Artifact Resolution profile），而这将利用比如 SOAP 映射产生一个回叫消息给应答实体以求重新获得<LogoutResponse>应答消息。

推荐将本步中的 HTTP 交换应用在安全传输层协议 TLS 1.0 之上以求保持机密性和消息的完整性。如果使用了 HTTP POST 或 Redirect 映射，则<LogoutRequest>注销请求消息需要被签字。如果使用了 HTTP Artifact 绑定，当重建被废弃时，还会提供一个鉴别请求发起者的替代方法。

第 11.4.4.4.2 节中有关于<LogoutResponse>应答消息内容的特定协议子集规则。

11.4.4.3.5 身份提供者发布<LogoutResponse>应答消息给会话参与者

如在前一步中所述，处理过最初发布注销请求的会话参与者的<LogoutRequest>注销请求消息之后，身份提供者必须回复一个<LogoutResponse>应答消息给源请求端，而且这个应答消息需包含能完成 SAML 协议交换的适当状态码。

将注销应答消息发送给最初注销请求发布者——会话参与者的过程利用了 SAML 绑定，这样的目的是为了与源请求端使用的映射相一致、与响应端的能力相一致、与身份提供者处用户代理的有效性相一致。假如第 1 步中使用了异步映射，那么被双方实体均支持的任何映射都可能会被使用。

第 11.4.4.4.2 节中有关于<LogoutResponse>应答消息内容的特定协议子集规则。

11.4.4.4 单点注销协议的使用

本节讲述<LogoutRequest>和<LogoutResponse>的用法。

11.4.4.4.1 <LogoutRequest>注销请求的用法

必须有<Issuer>发布者元素并且必须包含请求端实体的唯一标识符；属性的格式将会被忽略或者有属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

请求发布者必须已经通过了响应端的身份认证并且保证消息的完整性，通过对消息进行签字实现或者利用特定的映射机制实现均可。

请求中利用标识符对注销请求的责任人进行认证，这个标识符与请求发布者发布或接收的认证断言中的标识符高度匹配，此处的认证断言属于将被终结的会话，而每一种匹配规则都在第 8.2.7 节中有定义。

如果注销请求发布者是会话参与者，那么在请求中必须含有至少一个<SessionIndex>元素。如果请求发布者是会话管理者（或者按自己的要求运作），那么它将忽略任何起以下作用的元素——即表明所有与最初注销请求发起者相关会话的终结的元素。

注（资料性的）— PE38（见OASIS PE:2006）对以上段落进行了以下阐述：

如果注销请求发布者是会话参与者，那么在请求中必须含有至少一个<SessionIndex>元素（从第11.4节看，会话参与者常会收到用来发起会话的<saml:AuthnStatement>元素，此元素中包含SessionIndex属性）。

如果请求发布者是会话管理者（或者按自己的要求运作），那么它将忽略任何起以下作用的元素——即表明所有与最初注销请求发起者相关会话的终结的元素。

11.4.4.4.2 <LogoutResponse>注销应答的用法

必须有<Issuer>发布者元素并且必须包含应答实体的唯一标识符；属性的格式将会被忽略或者有属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

响应端必须已经通过了请求发布者的身份认证并且保证消息的完整性，通过对消息进行签字实现或者利用特定的映射机制实现均可。

11.4.4.5 元数据的使用

端点元素<md:SingleLogoutService>描述了所支持的映射和实体利用本规范发送请求和应答的可能路径。如果加密了源注销请求发起者的标识符，发起者可以使用响应端的<md:KeyDescriptor>元素，此元素有一个加密技术的使用属性可用来确定恰当的加密法则和设置以供使用，同时还有一个用于传送大量密钥的公钥。

11.4.5 名称标识符管理规范

在名称标识符管理规范支持的情境之中，身份提供者与服务提供商交换了一些源请求发起者固有标识符的格式，允许它们在一定时间内共用一些统一的标识符。随后，身份提供者可能会想要通知服务提供商格式和属性值的改变，格式和属性值将会用来识别同样的注销请求发布者。另一种选择是，服务提供商可能会想要为请求发布者配上它自己的别名"alias"，为了确保以后身份提供者与它进行关于请求发布者的通信时会包含这个别名。最后，服务提供商中的一个可能会想要去通知其他的知道，已经不再会利用一个特定标识符来发布或者接收消息。在这些情境中使用了 SAML 名称标识符管理协议协议子集。

注（资料性的）— PE12（见OASIS PE:2006）建议重写上面段落中的第二句如下：

随后，身份提供者可能会想要通知服务提供商属性值的改变，此属性值将会用来识别同样的注销请求发布者。

协议子集允许将协议与同步映射结合在一起，比如 SOAP 映射；或者与异步"front-channel"映射结合在一起，比如 HTTP Redirect、POST 或 Artifact 映射。front-channel 映射可能是必需的，比如为了实现改变需要在用户代理和应答提供者之间进行直接交互的情况。

11.4.5.1 必需的信息

标识： urn:oasis:names:tc:SAML:2.0:profiles:SSO:nameid-mgmt

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

11.4.5.2 协议子集概述

图 11-4 图示的是名称标识符管理协议子集的基本模版。

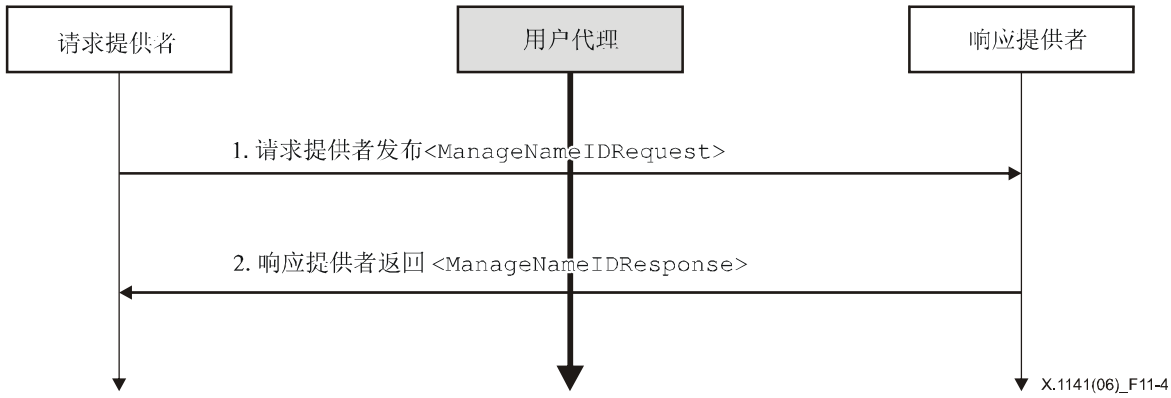

图 11-4/X.1141－名称标识符管理协议子集

灰度显示的用户代理表明消息的交换可能会通过用户代理或者是在系统实体间的直接交换，这一过程依赖于用来贯彻协议子集的 SAML 绑定。

协议子集还描述了以下的步骤。在一个单独的步骤中，可能会有一个或者更多基于本步骤中所使用映射的实际消息交换和其他的从属执行的行为：

1） **请求身份提供者/服务提供商发布<ManageNameIDRequest>**

步骤1中，身份提供者或服务提供商想要通知另一个提供者情况的改变，它会通过发送一个<ManageNameIDRequest>请求消息给此提供者来发起这个协议子集。这个请求消息可能会被直接传送给应答提供者或者是通过用户代理间接传送。

2） **响应身份提供者/服务提供商发布<ManageNameIDResponse>**

步骤2中，应答提供者（处理过请求之后）发布一个<ManageNameIDResponse>应答消息给源请求提供者。应答消息可能会被直接返回给请求提供者或者是通过用户代理间接返回（如果与步骤1中的请求表单保持一致）。

11.4.5.3 协议子集描述

下面的描述中提到了以下内容：

名称标识符管理服务

这是名称标识符管理协议在身份提供者或服务提供商处的端点，且这个身份提供者或服务提供商是<ManageNameIDRequest>或<ManageNameIDResponse>消息被交付的地方。请求和应答可以使用相同的端点或者不同的。

11.4.5.3.1 请求身份提供者/服务提供商发布<ManageNameIDRequest>

为了发起这个协议子集，请求发起者发布<ManageNameIDRequest>消息给另一个提供者的名称标识符管理服务请求端点。元数据可用做确定端点的路径和应答提供者支持的映射。

— **同步映射（Back-channel）**

请求发起者可以使用同步映射来直接发送请求给其他的提供者，比如 SOAP binding（见第 10 节）。请求发起者必须已经通过了其他提供者的身份验证，这可以通过签字请求消息<ManageNameIDRequest>实现也可以通过利用所支持的其他映射机制实现。

— **异步机制（Front-channel）**

另一种选择是，请求发起者可以使用（如果发起者有用户代理）一个异步机制来通过用户代理间接的发送请求给其他的提供者，比如 HTTP Redirect、POST 或 Artifact bindings（见第 10 节）。

如果使用了 HTTP Redirect 或 POST 映射，则<ManageNameIDRequest>消息在本步中被交付给其他的提供者。如果使用的是 HTTP Artifact 绑定，则其他的提供者使用第 11.4.6 节中定义的重建决定规范（Artifact Resolution profile），这将利用比如 SOAP binding 产生一个回叫信号给请求发起者以求能重新获得<ManageNameIDRequest>消息。

建议将本步中的 HTTP 交换应用在 TLS1.0（安全传输层协议）上以保持机密性和消息完整性。如果使用了 HTTP POST 或 Redirect 映射，则必须签字<ManageNameIDRequest>请求消息。如果使用 HTTP Artifact 绑定，则当消息的重建被摒弃时，会提供一个备用的认证请求发布者的方法。

每一个映射都会提供一个 RelayState 机制，请求发起者可以使用这个机制来联合协议子集交换和源请求。请求发起者必须在 RelayState 属性中暴露尽可能少的信息，除非使用的协议子集没有安全尺度的要求。

第 11.4.5.4.1 节中含有<ManageNameIDRequest>消息内容的协议子集特定规则。

11.4.5.3.2 响应身份提供者/服务提供商发布<ManageNameIDResponse>

接收端必须处理<ManageNameIDRequest>消息。处理过此消息或者遇到错误后，接收端需要发布包含适当状态码的<ManageNameIDResponse>应答消息给请求发布者来完成 SAML 协议交换。

— **同步映射（Back-Channel）**

如果请求发起者使用了同步映射，比如 SOAP binding（见第 10 节），则应答将被直接返回以完成同步通信。响应端必须已经通过了请求发布者的身份认证，这可以通过签字请求消息<ManageNameIDRequest>实现也可以通过使用所支持的其他映射机制实现。

— **异步映射（Front-channel）**

如果请求发起者使用的是异步映射，比如 HTTP Redirect、POST 或 Artifact bindings（见第 10 节），则应答<ManageNameIDResponse>（或其重建）将被通过用户代理间接返回给请求发起者的名称标识符管理服务应答端点。元数据可用于确定这个端点的路径和请求发起者支持的映射。这里可以使用任何两方实体均支持的映射。

如果使用了 HTTP Redirect 或 POST 映射，则<ManageNameIDResponse>消息在本步中被交付给请求发起者。如果使用的是 HTTP Artifact 绑定，则请求发起者使用第 11.4.6 节中定义的重建决定规范（Artifact Resolution profile），这将利用比如 SOAP binding 产生一个回叫信号给应答发起者以求能重新获得<ManageNameIDResponse>消息。

建议将本步中的 HTTP 交换应用在 TLS1.0（安全传输层协议）上以保持机密性和消息完整性。如果使用了 HTTP POST 或 Redirect 映射，则必须签字<ManageNameIDResponse>请求消息。如果使用 HTTP Artifact 绑定，则当消息的重建被摒弃时，会提供一个备用的认证应答发布者的方法。

第 11.4.5.4.2 节中含有<ManageNameIDResponse>消息内容的协议子集特殊规则。

11.4.5.4 名称标识符管理协议的使用

本段包括 ManageNameIDRequest 和 ManageNameIDResponse 的使用方法。

11.4.5.4.1 <ManageNameIDRequest>的使用方法

必须具备<Issuer>元素且含有请求实体的唯一标识符，必须忽略它的特性格式或者有一个属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

请求发起者必须通过了响应端的身份认证并且确保消息完整性，这可通过签字消息实现或者利用特定的映射机制实现。

11.4.5.4.2 <ManageNameIDResponse>的用法

必须具备<Issuer>元素且含有应答实体的唯一标识符，必须忽略它的特性格式或者有一个属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

响应端必须通过了请求发起者的身份认证并且确保消息完整性，这可通过签字消息实现或者利用特定的映射机制实现。

11.4.5.5 元数据的使用

端点元素<md:ManageNameIDService>描述了支持的映射和实体利用本协议子集发送请求和应答的路径。如果加密了责任人的标识符，则请求发起者可以使用响应端的<md:KeyDescriptor>元素，此元素带有加密使用特性可用来确定恰当的加密算法和设置，以及在传送大量密钥时使用的公钥。

11.4.6 重建决定规范

第 10 节定义了针对摒弃 SAML 重建转向相应协议消息的重建决定协议。HTTP Artifact 绑定（见第 10 节）在传递所涉及的 SAML 协议消息机制中起杠杆作用。这个协议子集描述了带有同步映射的协议使用，比如第 10 节中定义的 SOAP 映射。

11.4.6.1 必需的信息

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

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

11.4.6.2 协议子集概述

第 8 节中定义了管理本协议子集的消息交换规则和基本处理规则，包括要被交换的消息以及用于交换消息的映射。本建议书的第 10 节定义了 SOAP V1.1 消息交换的映射。除非建议中特别指明，不然所有的请求都定义在这些规范说明之中。

图 11-5 图示阐明了重建决定协议子集的基本模版。

图 11-5/X.1141－重建决定协议子集的基本模版

协议子集描述如下：

1） 请求实体发布<ArtifactResolve>

步骤1中，请求发起者通过发送<ArtifactResolve>消息给重建发布者来发起这个协议子集。

2） 应答实体发布<ArtifactResponse>

步骤2中，响应端（处理过请求之后）发布一个<ArtifactResponse>消息给请求端。

11.4.6.3 协议子集描述

下面的描述中涉及：

— **重建决定服务（Artifact resolution service）**

这是重建发布者处的重建决定协议端点，<ArtifactResolve>消息将被交付到这里。

11.4.6.3.1 请求实体发布<ArtifactResolve>消息

为了发起协议子集，请求端收到重建要求和利用 SourceID 确定发布者之后发送一个含有此重建要求的<ArtifactResolve>消息给重建发布者的重建决定服务端点。元数据可用于确定这个端点的路径和重建发布者支持的映射。

请求端需使用同步映射直接发送请求给重建发布者，比如 SOAP binding（见第 10 节）。请求端需要通过响应端的身份认证，这可通过签字<ArtifactResolve>消息实现或者利用所支持的其他映射机制。使用 HTTP Artifact 绑定的特定协议子集可能会有额外的需求，这样身份认证也是强制的。

第 11.4.6.4.1 节中含有<ArtifactResolve>消息内容的特定协议子集规则。

11.4.6.3.2 响应实体发布<ArtifactResponse>

重建发布者根据第 8 节中定义的那样处理<ArtifactResolve>消息。处理过请求消息或遇到错误之后，重建发布者需要返回一个含有恰当状态码的<ArtifactResponse>应答消息给请求端来完成此 SAML 协议交换。如果成功，那么摒弃的 SAML 协议消息与其相应的重建一起都会被使用。

响应端需要通过请求端的身份认证，这可通过签字<ArtifactResponse>消息实现或者利用所支持的其他映射机制。

第 11.4.6.4.2 节中含有<ArtifactResponse>消息内容的特定协议子集规则。

11.4.6.4 重建决定协议的使用

本段包括 ArtifactResolve 和 ArtifactResponse 的使用方法。

11.4.6.4.1 <ArtifactResolve>消息的用法

必须具备<Issuer>元素且包含请求实体的唯一标识符，必须忽略格式属性或者含有一个属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

请求端需要通过响应端的身份认证并且保证消息的完整性，这既可以通过对消息签字实现也可以利用特定的映射机制。使用 HTTP Artifact 绑定的特定协议子集可能会有额外的需求，这样身份认证也是强制的。

11.4.6.4.2 <ArtifactResponse>应答消息的用法

必须具备<Issuer>元素且包含重建发布者的唯一标识符，必须忽略格式属性或者含有一个属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

响应端需要通过请求端的身份认证并且保证消息的完整性，这既可以通过对消息签字实现也可以利用特定的映射机制。

11.4.6.5 元数据的使用

第 9 节定义了一个索引端点元素<md:ArtifactResolutionService>，用来描述所支持的映射和请求端利用本协议子集发送请求消息的可能路径。这个索引特性用于区分那些可能会在重建消息的 EndpointIndex 域中提及的端点。

11.4.7 断言查询/请求协议子集

第 10 节中定义了用于请求现有断言的协议，这个请求通过参考信息或者通过以主体和额外的特定断言标准为基础的查询来实现。这个协议子集描述了本协议针对同步映射的使用，比如第 10 节中定义的 SOAP 映射。

11.4.7.1 必需的信息

标识：urn:oasis:names:tc:SAML:2.0:profiles:query

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

11.4.7.2 协议子集概述

第 8 节中定义了管理本协议子集的消息交换规则和基本处理规则，包括要被交换的消息以及用于交换消息的映射。本建议书的第 10 节定义了 SOAP V1.1 消息交换的映射。除非建议中特别指明，不然所有的请求都定义在这些规范说明之中。

图 11-6 图示阐明了查询/请求协议子集的基本模版。

图 11-6/X.1141－查询/请求协议子集（query/request profile）的基本模版

协议子集描述了以下几步：

1） SAML请求发布者发布查询/请求（Query/request）

第 1 步中，SAML 请求端通过发送 <AssertionIDRequest>、<SubjectQuery>、<AuthnQuery>、<AttributeQuery>或<AuthzDecisionQuery>消息给SAML管理者来发起这个协议子集。

2） SAML管理者发布<Response>应答

第2步中，应答SAML管理者（处理过查询或请求之后）发布<Response>应答消息给SAML请求端。

11.4.7.3 协议子集描述

下面的描述包括：

— **查询/请求服务（Query/Request Service）**

这是位于 SAML 管理者处的查询/请求协议端点，查询或<AssertionIDRequest>消息将被交付到这里。

11.4.7.3.1 SAML请求端发布查询/请求

为了发起这个协议子集，SAML 请求端发布 <AssertionIDRequest>、<SubjectQuery>、<AuthnQuery>、<AttributeQuery>或<AuthzDecisionQuery>消息给 SAML 管理者的查询/请求服务端点。元数据可以用做确定这个端点的路径和 SAML 管理者支持的映射。

SAML 请求端使用同步映射来直接发送请求给身份提供者，比如 SOAP binding（见第 10 节）。请求端需要通过 SAML 管理者的身份认证，这可通过对消息签字实现或者利用所支持的其他映射机制。

第 11.4.7.4.1 节中有关于各种消息内容的特定协议子集规则。

11.4.7.3.2 SAML管理者发布<Response>应答

SAML 管理者必须按照第 8 节中定义的那样处理查询或请求消息。处理过消息或者遇到错误之后，SAML 管理者必须返回一个包含适当状态码的<Response>应答消息给 SAML 请求端来完成此 SAML 协议交换。如果请求成功地找到了一个或多个匹配的断言，则会包括在应答之中。

响应端需要通过请求端的身份认证，这可通过对<Response>应答消息签字实现或者利用所支持的其他映射机制。

第 11.4.7.4.2 节中有关于<Response>消息内容的特定协议子集规则。

11.4.7.4 查询/请求协议（Query/Request Protocol）的使用

本段定义了 SAML 管理者的查询/请求协议端点，查询消息将被交付到这里。

11.4.7.4.1 查询/请求的用法

必须具备<Issuer>元素。

请求端需要通过响应端的身份认证并且保证消息的完整性，这可通过对消息签字实现或者利用特定的映射机制。

11.4.7.4.2 <Response>的用法

必须具备<Issuer>元素并且<Issuer>元素要包含做出应答的 SAML 管理者的唯一标识符；属性格式可忽略或者有属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。这并不需要与返回的断言之中的<Issuer>元素匹配。

响应端需要通过请求端的身份认证并且确保消息的完整性，这可通过对消息签字实现或者是利用特定的映射机制。

11.4.7.5 元数据的使用

第 9 节定义了几个端点元素，<md:AssertionIDRequestService>、<md:AuthnQueryService>、<md:AttributeService>和<md:AuthzService>，用来描述所支持的映射和请求端利用本协议子集发送请求或查询的路径。

如果加密最终断言或者针对特定实体的断言内容，则 SAML 管理者可以使用此实体的带有加密使用特性的<md:KeyDescriptor>元素来确定一种恰当的加密算法和设置，包括在传送大量密钥中使用的公钥。

不同的任务描述符可能含有<md:NameIDFormat>、<md:AttributeProfile>和<saml:Attribute>元素来指明支持特定名称标识符格式、特性规范或特殊属性和值的能力。在给定请求中支持任何这些要求的能力都依靠管理者的决策和判断力。

11.4.8 名称标识符映射协议子集

第 8.2.6 节段定义了名称标识符映射协议，用于将同一责任人的名称标识符映射成不同的名称标识符。本协议子集描述了此协议结合同步映射的使用，比如第 10 节中定义的 SOAP 映射，以及利用加密方法保护责任人的机密性和限制映射标识符的使用的其他策略。

11.4.8.1 必需的信息

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

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

11.4.8.2 协议子集概述

第 8 节中定义了管理本协议子集的消息交换和基本处理的规则，还定义了将被交换的消息以及用于交换消息的映射。本建议书的第 10 节定义了 SOAP V1.1 消息交换的映射。除非特别指明，所有的需求都定义在这些规范说明之中。

图 11-7 图示阐明了名称标识符映射协议子集的基本模版。

图 11-7/X.1141－名称标识符协议子集基本模版

协议子集描述了以下几步：

1） **请求实体发布<NameIDMappingRequest>请求消息**
在第1步中，请求端通过发送<NameIDMappingRequest>消息给一个身份提供者来发起此协议子集。

2） **身份提供者发布<NameIDMappingResponse>应答消息**
第2步中，发布应答的身份提供者（处理过请求之后）发布一个<NameIDMappingResponse>应答消息给请求端。

11.4.8.3 协议子集描述

本子段使用名称标识符映射服务（Name Identifier Mapping Service），这是位于身份提供者处的名称标识符映射协议端点，<NameIDMappingRequest>请求消息将交付于此。

11.4.8.3.1 请求实体发布<NameIDMappingRequest>请求消息

为了发起此协议子集，请求端发布<NameIDMappingRequest>消息给身份提供者的名称标识符映射服务端点。元数据可用来确定此端点的路径和身份提供者所支持的映射。

请求端必须使用同步映射来直接发送请求给身份提供者，比如 SOAP 映射（见第 10 节）。请求端需要通过身份提供者的身份认证，这可通过对<NameIDMappingRequest>消息签字实现或者利用所支持的其他映射机制。

第 11.4.8.4.1 节中包括了<NameIDMappingRequest>消息内容的特定协议子集规则。

11.4.8.3.2 身份提供者发布<NameIDMappingResponse>应答消息

身份提供者必须依照第 8 节中定义的那样处理<ManageNameIDRequest>请求消息。处理过消息或遇到错误之后，身份提供者必须返回一个包含恰当状态码的<NameIDMappingResponse>应答消息给请求端来完成 SAML 协议交换。

响应端需要通过请求端的身份认证，这可通过对<NameIDMappingResponse>应答消息签字实现或者利用所支持的其他映射机制。

第 11.4.8.4.2 节中包括了<NameIDMappingResponse>消息内容的特定协议子集规则。

11.4.8.4 名称标识符映射协议的使用

第 8 节中定义了名称标识符映射协议，此协议用来将同一个责任人的名称标识符映射为不同的名称标识符。本段描述了此协议的使用和用于保护责任人机密性的附加策略，比如限制映射标识符的使用。

11.4.8.4.1 <NameIDMappingRequest>请求消息的用法

必须具备<Issuer>元素。

请求端需要通过响应端的身份认证并且保证消息的完整性，这可通过对消息进行签字实现或者利用特定的映射机制。

11.4.8.4.2 <NameIDMappingResponse>应答消息的用法

必须具备<Issuer>元素并且此元素包含响应端身份提供者的唯一标识符；属性的格式可忽略或者是有属性值：urn:oasis:names:tc:SAML:2.0:nameid-format:entity。

响应端需要通过请求端的身份认证并且保证消息的完整性，这可通过对消息进行签字实现或者利用特定的映射机制。

W3C Encryption:2002，第 2.2.3 节，定义了用于名称标识符机密性的加密法的使用。在大多数情况下，身份提供者应该加密它所返回给请求端的映射名称标识符，以保护责任人的机密性。请求端能够提取出<EncryptedID>元素并将它放置到随后的协议消息或断言之中。

映射标识符的限制使用

身份提供者会对结果标识符的使用做额外的限制，此限制通过以<Assertion>断言消息的形式返回映射名称标识符来实现，而这个<Assertion>断言消息在它的<Subject>中包含标识符但是没有任何说明。接下来此断言消息被加密然后作为<EncryptedID>中的<EncryptedData>元素返回给请求端。断言消息中可以包含<Conditions>元素来限制使用，如同第 8 节中定义的那样，比如基于时间的约束或者特定依赖用户的使用，并且为了保护完整性此消息必须被签字。

11.4.8.5 元数据的使用

本节定义了一个端点元素，<md:NameIDMappingService>，来描述支持的映射和请求端利用此协议子集发送请求的可能路径。

如果将特定实体的结果标识符加密，则身份提供者可以使用此实体的带有加密法可用属性的<md:KeyDescriptor>元素来确定一个恰当的加密算法和设置，包括在传送大量密钥的过程中使用的公钥。

11.4.9 SAML特性协议子集

当处理特性信息的特殊类型时或者当与要求更严格的外部系统有互作用时，特性协议子集提供了约束 SAML 特性表述所必需的定义。本子段指定了 SAML 的基本特性协议子集，X>500/LDAP 协议子集、UUID 协议子集以及 XACML 协议子集。

11.4.9.1 基本特性协议子集

基本特性协议子集的指定是单一进行的，但并不唯一，它的命名以 SAML 特性为基础，此 SAML 特性与基于内置在 W3C 中的 Datatypes 数据类型的属性值结合在一起，排除了对可使句法有效的扩展 Schema 的需求。

必需的信息

标识：urn:oasis:names:tc:SAML:2.0:profiles:attribute:basic

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

SAML 属性性命名

在<Attribute>属性元素中的名称格式 XML（NameFormat XML）属性值必须是 urn:oasis:names:tc:SAML:2.0:attrname-format:basic。

Name XML 属性必须与此格式的特定规则结合在一起，如同第 8 节中定义的那样。

— **属性名称对比**

当且仅当两个<Attribute>元素的 Name XML 属性相同时（依据第 8 节中描述的判断），这两个元素才代表同样的 SAML 属性。

XML 属性的特定协议子集

没有额外定义与<Attribute>元素一起使用的 XML 属性。

SAML 属性值

<Attribute Value>元素内容的预期格式必须是附件 A 中所定义类型中的一种。必须具备属性 xsi:type 并且赋予它适当的值。

举例

```
<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>
```

11.4.9.2 X.500/LDAP属性协议子集

基于一系列 ITU-T X.500 建议书和 IETF RFC 3377 的目录被广泛地使用。目录 Schema（Directory schema）用来模拟将被存放在这些目录中的信息。特别是在 X.500 中，属性类型的定义，作为目录中的基本信息存储元素（本建议书中称它们为“directory attributes”），用来指明属性的句法规则和其他特征。目录属性类型的定义出现在 X.500 和 LDAP 说明书中的 schema 中或者其他公共文档（比如 inetOrgperson schema（见 IETF RFC 2798））的 schema 中或者是为个人目的所定义的 schema 之中。在任何情况之下，配置者利用这些位于 SAML 属性说明之中的目录属性类型是很有好处的，不需要手工为它们创建特定 SAML 属性定义，而只需以一个可共同使用的形式来完成。

当被表述为 SAML 属性时，X.500/LDAP 属性协议子集为这些属性的命名和表述定义了一个公共协定。

必需的信息

标识：urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500 （这也是位于附件 A 中的相应的 X.500/LDAP 协议子集 Schema 所分派的目标名称段。）

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

SAML 属性命名

<Attribute> 中的 NameFormat XML 属性为：urn:oasis:names:tc:SAML:2.0:attrname-format:uri。

为了构造属性名称，使用了 IETF RFC 3061 中描述的 URN oid 名称段。在这个方法之中，Name XML 属性是基于被分配给目录属性类型的客体标识符。

例如：

urn:oid:2.5.4.3

由于 X.500 程序要求每一个属性类型都标识为一个唯一的客体标识符，此命名模式确保得来的 SAML 属性名称是明确不模糊的。

为了增强可读性，还会有针对于携带一个与 OID URN（如同 IETF RFC 3061 中所定义的）一起的可选字符串名称这样的应用的要求。可选择的 XML 属性特殊体（第 8 节中定义）就是为这个目的而存在的。如果目录属性类型的定义包含一个或者多个属性类型的描述符（短名称），那么如果存在 FriendlyName 属性值的话，此属性值就是已定义的描述符中的一个。

当且仅当两个<Attribute>元素的 Name XML 属性值相同时（依据 IETF RFC 3061 中的判断），这两个元素才代表同一个 SAML 属性。FriendlyName 属性不参与比较。

特定协议子集 XML 属性

没有额外定义与<Attribute>元素一起使用的 XML 属性。

SAML 属性的值

用在本地 X.500 目录中的目录属性类型定义利用 ASN.1 指明了属性的句法规则。为了在 LDAP 中使用，目录属性的定义额外包括了 LDAP 句法规则，此句法规则指明了属性或断言的值怎样与因在 LDAP 协议中传输而呈现的句法规则保持一致（即 LDAP 特殊编码）。LDAP 特殊编码以 UTF-8 的形式产生 Unicode 字符。此 SAML 属性协议子集仅仅为具备 LDAP 句法规则的那些目录属性指明了 SAML 属性值的格式。将来对此协议子集的扩展可以那些句法规则具备其他编码方法的目录属性定义属性值的格式。

为了重现特殊属性值的编码规则，<AttributeValue>元素必须含有一个在 XML 名称段中定义的 XML 属性指定的 Encoding，urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500。

对于任何其句法规则的 LDAP 特殊编码专门产生 UTF-8 字符串做值的目录属性来讲，SAML 属性值被简单编码为 UTF-8 字符串，如同<AttributeValue>元素的内容，没有任何额外的空白地带。在这种情况下，XML 属性 xsi:type 必须设置为 **xs:string**。而且会为协议子集特殊编码 XML 属性提供一个“LDAP”属性值。

一些 LDAP 属性句法（结合 OID）如下：

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

对于其他所有的 LDAP 句法来讲，其属性值编码使用基 64 编解码围绕 ASN.1 八位字节串编码 LDAP 属性值，如同<AttributeValue>元素的内容。XML 属性 xsi:type 必须设置为 **xs:base64Binary**。而且会为协议子集特殊编码 XML 属性提供一个“LDAP”属性值。

当比较 SAML 属性值的等同性时，必须观测为相应的目录属性类型指定的匹配规则（比如场景灵敏度）。

特定协议子集的 Schema

下面的 Schema 列表表明了如何定义特定协议子集的 Encoding XML 属性的（见附件 A）：

```
<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 identity: 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 属性协议子集, first published in SAML
2.0.
        </documentation>
    </annotation>
    <attribute name="Encoding" type="string"/>
</schema>
```

例如

下面是一个目录属性"givenName"的映射举例，重现了 SAML 断言主体的名称。它的目标标识符是{joint-iso-itu-t(2) ds(5) attributeType(4) givenName(42)}，LDAP 句法是 Directory String。

```
<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>
```

11.4.9.3 UUID属性协议子集

UUID 属性协议子集将 UUID 属性值的表述标准化为 SAML 属性名称和数值。当属性的源系统是利用 UUID 识别一个属性或其值时，上述方法是适用的。

必需的信息

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

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

SAML 属性命名

<Attribute>元素中的 NameFormat XML 属性：urn:oasis:names:tc:SAML:2.0:attrname-format:uri

如果下面对属性名称的表述是 UUID，那么就使用在 ITU-T Rec. X.667 中描述的 URN uuid 名称段。这种方法中，Name XML 属性是基于下面识别属性的 UUID 的 URN 格式。

例如：

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

如果下面对属性名称的表述不是 UUID，那么 Name XML 属性就可以使用任何形式的 URI。

为了增强可读性，还会有针对于—携带一个与 URI 一起的可选字符串名称—这样的应用的要求。可选的 XML 属性 FriendlyName 就是为了这个目的而存在的。

当且仅当两个<Attribute>元素的 Name XML 属性值相同时-依据 ITU-T Rec. X.667 判断，它们表示的才是同一个 SAML 属性。FriendlyName 属性不参与对比。

特定协议子集 XML 属性

没有额外定义与<Attribute>元素一起使用的 XML 属性。

SAML 属性值

在属性值仍为 UUID 的情况下，上面描述的同一个 URN 句法用来在<AttributeValue>元素中表示属性值。XML 属性 xsi:type 必须设置为 **xs:anyURI**。

如果属性的值不是 UUID，那么对<AttributeValue>元素的使用就没有限制。

举例

下面是一个 DCE Extended Registry Attribute（DCE 扩展注册属性）的例子，"pre_auth_req"的设置，有众所周知的 UUID：6c9d0ec8-dd2d-11cc-abdd-080009353559，且是整数值。

```
<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>
```

11.4.9.4 XACML属性协议子集

依照 ITU-T Rec. X.1142，可使用 SAML 属性断言作为授权决定的输入。由于 SAML 属性格式与 XACML 属性格式不同，就必须执行一次映射。XACML 属性协议子集利用标准化的命名、属性值的句法和额外的属性元数据来推动这次映射。与此协议子集一起产生的 SAML 属性能自动被映射为 XACML 属性，并作为 XACML 授权决定的输入。

必需的信息

标识：urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML（这也是附件 A 中相应的 XACML 协议子集所指定的目标名称段。

联系方式：security-services-comment@lists.oasis-open.org

说明：见下文。

更新：无。

SAML 属性命名

<Attribute>元素中 NameFormat XML 属性：urn:oasis:names:tc:SAML:2.0:attrname-format:uri。

如同第 8 节中定义的那样，Name XML 属性必须遵循为此格式而定的规则。

为了增强可读性，还会有针对于—携带一个与 OID URN 一起的可选字符串名称—这样的应用的要求。可选择的 XML 属性 FriendlyName（第 8 中定义的）就是为此目的而存在的，但是不能移植到与之相当的 XACML 属性之中。

当且仅当两个<Attribute>元素的 Name XML 属性值在二进位的比较下等同时，它们表示的才是同一个 SAML 属性。FriendlyName 属性不参与比较。

特定协议子集 XML 属性

XCAML 要求每一个属性都带有一个外在的数据类型。为了提供这个数据类型的值，在 XML 名称段 urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML 中，定义了一个新的 URI-valued XML 属性叫做 DataType。

与此协议子集一致的 SAML <Attribute>元素必须包含有效名称段 DataType 属性，或者属性值假定为：http://www.w3.org/2001/XMLSchema#string。

如果使用非标准的属性值，那么每一个会强制用非标准的 DataType 属性值映射 SAML 属性的 XACML PDP 必须扩展支持新的数据类型。

SAML 属性值

<AttributeValue>元素内容的句法格式必须与在上一级<Attribute>元素中出现的特定协议子集 DataType XML 属性中的数据类型保持一致。为了数据类型与第 8 节中定义的类型一致性，xsi:type XML 属性也必须被用于<AttributeValue>元素。

特定协议子集 Schema

下面的 Schema 列表表示的是怎样定义 profile-specific `DataType` XML 属性的（附件 A）：

```
<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 identity: 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>
```

例如

下面是一个映射"givenName" LDAP/X.500 属性的例子，重现了 SAML 断言主体的名称。它还表明当彼此兼容时，一个单独的 SAML 属性可以与多个属性协议子集保持一致。

```
<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>
```

注（资料性的）— PE39（见OASIS PE:2006）对上例的阐述如下：

```
<saml:Attribute
  xmlns:xacmlprof="urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML"
      xmlns:ldapprof="urn:oasis:names:tc:SAML:2.0:profiles: AttributeValue :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>
```

12 SAML认证关联

本建议定义了一个认证关联断言的定义的句法以及一个认证关联类别的初始列表。

12.1 认证关联概念

如果信任方依赖于权威认证机构对主体的认证，那么为了评估它在断言中能设置的信任级别，该信任方还需要除了断言本身以外的额外信息。本建议定义了用于生成认证关联断言的 XMLSchema，也就是允许认证授权中心向该信任方提供此额外信息的 XML 文本。此外，本建议还定义了一些认证关联类别，所有认证关联断言将按这些类别进行分类，以此来简化它们的解释。

SAML 并没有规定有关认证授权中心向主体分配 ID 标识的过程以及那些主体随后到认证授权中心去认证的过程的单个技术、协议或策略。不同的认证授权中心对于如何认证主体将会选择不同的技术，采用不同的程序，并被不同的合法职责所限制。

认证授权中心的选择在很大程度上受与认证授权中心交互的信任方的需求影响。这些需求本身由信任方向主体提供的业务（也就是任何信息交换的灵敏度、相关的财政值、信任方的风险承受度等）所决定。

因此，如果信任方在它从认证授权中心接收到的认证断言中具有足够的信任级别，那么对于除了价值不高的业务之外的任何事情将有必要知道初始认证机制使用何种技术、协议和程序，认证断言就是基于该初始认证机制的。用该信息武装起来并信任实际断言的起源，信任方在考虑应该允许认证断言中的何种业务主体接入时将能更好的做出决定。

认证关联定义为信任方在做出有关认证断言的权限决定之前可能会需要的除了认证断言本身之外的信息。该关联可能包括但不限于所使用的实际的认证方法。

12.2 认证关联断言

如果信任方依赖于认证授权中心对另外一个实体的认证，那么为能够将认证加入到一个风险关联，该信任方可能会需要除了认证本身之外的其他信息。这些信息包括：

- 初始化用户标识机制（例如：面对面、在线、共享密钥）。
- 信任状（例如，信任刷新频率、客户端Key值的产生）的最小组合机制。
- 存储和保护信任状（例如：智能卡、密码规则）的机制。
- 认证机制或方法（例如：密码的方式）。

上面特征列表的变更和排列保证所有的认证断言不会相同；特定的认证断言将会由每个这些（或其他）变量的值来区分。

一个 SAML 认证授权中心能够以认证关联断言的形式向信任方传送额外的认证关联信息，一个 XML 文本或者直接插入，或者在认证授权中心向信任方提供的认证断言中引用。

SAML 请求端能够通过在认证请求中识别特定的认证关联来要求一个认证按照该认证关联执行。请求端也可以指定该认证必须由一个超过设定值的（在允许范围内的超过值）认证关联引导。

12.2.1 数据模型

本建议定义了一个特别的认证关联断言包含方法、程序和认证授权中心在发布一个 ID 标识之前用来检验该主体的机制，保护后续认证所需要的保密信息，以及用于本次认证的机制等特征。这些特征在认证关联 Schema 中按下列方式分类：

- 辨认类－那些描述方法和认证授权中心用来在一个主体和标识主体的ID（或名字）之间产生初始联系的机制的特征。
- 技术保护类－那些描述如何保证“保密消息”（该主体用于在认证授权中心处认证的知识或财产）安全的特征。
- 操作保护类－那些描述认证授权中心所使用的程序的安全控制（例如：安全审计、档案记录等）的特征。
- 认证方法类－那些提到的断言的主体用于向认证授权中心认证而定义的机制(例如：一张智能卡对应的密码)等特征。

- 政府许可类－那些描述在认证事件和/或与它相关的技术认证结构中潜在的法律框架的特征。

12.2.2 扩展性

认证关联断言 Schema 通过<Extension>元素很好的定义了扩展点。认证授权中心可以用此元素来为他们提到的 SAML 断言插入附加的认证关联细节信息（假设那些强烈的信任方能够识别和理解这些扩展）。这些附加元素必须在相对于认证关联断言基础或该断言本身所属类别 Schema 的一个独立的 XML 命名域中。

12.2.3 处理规则

第 8 节给出了认证关联断言的附加处理规则，这些处理规则调用了特定认证关联断言的相关强度或质量的公共解释，并且这些规则作为可选规则不能用绝对项表示。

12.2.4 Schema

本节属非标准化。

附录六给出了一个用于确认单个常规断言的完整的认证关联类别 XMLSchema 列表和认证关联 XMLSchema。

12.3 认证关联类别

不同特征变换的数量确保了独特的认证关联在理论上其数量是有限的。这意味着，在理论上，任何特定的信任方都能够解析任意的认证关联断言，而且更重要的是为评估相关认证断言的质量对该断言进行分析。这种评估是非同寻常的。

幸运的是，还可以进行优化。实际上许多认证关联将根据行业实践和技术来决定其所属的分类。目前，许多 B2C 的 Web 浏览器认证关联将（部分）由主体来定义，该主体经认证授权中心通过一个受 TLS 保护的会话密码对其进行认证。在企业界，基于证书的认证是很普遍的。当然，并非所有认证关联都局限于该主体如何通过认证等特征。然而，认证方法通常是最明显的特征，例如：为一类有关系的认证关联充当一个有用的分类器。

本建议中所表达的概念作为一系列认证关联类别的定义。每一类都定义了所有认证关联集的一个子集。这些类型被当做目前认证技术和实践的代表，并且当提到认证关联的话题时向断言方和信任方提供一个方便的速记。

举例说明，一个认证授权中心可能包括全部认证关联断言，它向信任方提供一个该认证关联也属于某个认证关联类别的断言。对于相关的认证断言为一些信任方分配适当的信任级别，该断言已经足够详细。其他信任方可能宁愿自己来检查所有的认证关联断言。同样地，指向一个认证关联类别而不是被要求列出指定认证关联断言的所有细节的能力将会简化信任方向一个认证授权中心表达它的设计和/或需求的过程。

12.3.1 认证关联类别的优势

类型附加层的介绍以及有代表性的和灵活的类型的初始列表定义如下：

- 通过给认证授权中心和信任方一个供讨论的框架，使得他们更容易在哪些是可接受的认证关联的问题上面达成协议。
- 使得信任方在向认证授权中心请求一个递升的认证断言时更容易做出参数选择。
- 通过给信任方归属哪种类型的选择权，从而减轻它们处理认证关联断言的负担。
- 将信任方从新认证技术的冲击中隔离开。

- 使得认证授权中心更容易公布他们的认证能力，例如，通过WSDL。

12.3.2 处理规则

第 8 节给出了对认证关联类别的进一步处理规则。在大多数情况下，这些扩展处理规则应共享调用特定认证关联类别的有关强度或质量的常规解释，并且不能以常数项表达，这些扩展的处理规则作为可选的规则来执行。

12.3.3 可扩展性

在核心的认证关联断言方案中，个别的认证关联类别方案遵守树型结构特定位置的<Extension>元素。通常情况下，<Extension>单位作为<xs:choice>元素的子元素出现，在生成作为基础类型限制的适当类型方案的定义时这个选项被删除。当<Extension>元素作为<xs:sequence>元素的可选子元素出现时，除了任何必须元素外<Extension>元素也允许保留。

因此，认证关联断言可以包括<Extension>元素（带有属于不同命名空间的附加元素）并且如果遇到其他有关 Schema 的需求仍然遵守认证关联类别 Schema。

认证关联类别 Schema 将类型定义限制在基础认证关联 Schema 内。作为一个扩展点，认证关联类别 Schema 本身被进一步限制，他们的类型定义在其他的 Schema 中充当基础类型（由一些团体希望有一个更严谨定义的认证关联类别而定义的）。为防止逻辑上产生矛盾，任何这种 Schema 的扩展仅可以进一步约束类型 Schema 的类型定义。为加强这一限制，认证关联类别 Schema 定义中规定<schema>元素中属性 finalDefault="extension"，以防止此类引用源。

12.3.4 Schema

在下面几小节中列出了有关认证关联类别。这些类型按字母顺序列出，这些类型的排列顺序没有任何其他含义。执行者可以根据本建议书中（第 13 节）给出的一致性指导来选择支持哪一个类型。这些类型由 URI 唯一标识，URI 的初始头规定如下：

```
urn:oasis:names:tc:SAML:2.0:ac:classes
```

类型 Schema 定义为基础认证关联“类型”Schema 的有限子集。据称，相对于一个给定的认证关联类别 Schema 而确定的 XML 实例符合该认证关联类别。

由于类型 Schema 输入类型 Schema 命名空间并在该空间中重定义了元素和类型，class-conforming 认证关联断言不会同时对基础认证关联 Schema 进行证实。

12.3.4.1 网际协议

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当主体仅使用提供的 IP 地址通过认证时，网际协议类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
  targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol
        Document identity: saml-schema-authn-context-ip-2.0
```

```
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="IPAddress"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.2 网际协议密码

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword`

注意，该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当主体使用提供的 IP 地址和用户名、密码通过认证时，此网际协议密码类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
  targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword"
  xmlns:ac="urn:oasis:names:tc:SAML:2.0:ac"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">
```

```
  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword
        Document identity: saml-schema-authn-context-ippword-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="IPAddress"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.3 Kerberos

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当主体为获得一个 Kerberos 入场券而使用密码向本地认证授权中心认证时，该类型可用。此 Kerberos 入场券将用于随后的网络认证。

注 1 — 当认证该主体时，认证授权中心有可能指出（通过Kerberos关联类型）一个Kerberos密钥分发中心[IETF RFC 1510]使用的预认证数据类型。认证授权中心获取该信息的方法不在本建议范围内，但是强烈建议使用一种可靠的方法向认证授权中心传送预认证数据类型和任何其他跟Kerberos相关的关联细节（例如，入场券的生命周期）。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos
        Document identity: saml-schema-authn-context-kerberos-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
              <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
```

```
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SharedSecretChallengeResponse"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SharedSecretChallengeResponseType">
      <xs:complexContent>
        <xs:restriction base="SharedSecretChallengeResponseType">
          <xs:attribute name="method" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

下面是符合该类型 Schema 的一个 XML 实例：

```
<AuthenticationContextDeclaration
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos">

  <AuthnMethod>

    <PrincipalAuthenticationMechanism preauth="0">
      <RestrictedPassword>
        <Length min="4"/>
      </RestrictedPassword>
    </PrincipalAuthenticationMechanism>

    <Authenticator>
      <AuthenticatorSequence>
        <SharedSecretChallengeResponse
method="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"/>
      </AuthenticatorSequence>
    </Authenticator>

  </AuthnMethod>

</AuthenticationContextDeclaration>
```

注 2 — 附录四介绍了SSL的使用。

12.3.4.4 MobileOneFactorUnregistered

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregistered`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

该类型反映出没用移动用户的注册过程，而且移动设备的认证没有要求端用户的交互。该关联类型仅认证设备，不认证用户，它在除了移动运营商希望在其认证过程中增加一个安全设备认证之外的业务中有用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema

targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnre
gistered"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"

xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregistered"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">
```

```
    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregistered
        Document identity:
saml-schema-authn-context-mobileonefactor-unreg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="ZeroKnowledge"/>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
              <xs:element ref="AsymmetricDecryption"/>
              <xs:element ref="AsymmetricKeyAgreement"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
```

```
              <xs:element ref="SSL"/>
              <xs:element ref="MobileNetworkNoEncryption"/>
              <xs:element ref="MobileNetworkRadioEncryption"/>
              <xs:element ref="MobileNetworkEndToEndEncryption"/>
              <xs:element ref="WTLS"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="OperationalProtectionType">
      <xs:complexContent>
        <xs:restriction base="OperationalProtectionType">
          <xs:sequence>
            <xs:element ref="SecurityAudit"/>
            <xs:element ref="DeactivationCallCenter"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TechnicalProtectionBaseType">
      <xs:complexContent>
        <xs:restriction base="TechnicalProtectionBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PrivateKeyProtection"/>
              <xs:element ref="SecretKeyProtection"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrivateKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="PrivateKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecretKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="SecretKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
```

```
                <xs:enumeration value="MobileDevice"/>
                <xs:enumeration value="MobileAuthCard"/>
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecurityAuditType">
      <xs:complexContent>
        <xs:restriction base="SecurityAuditType">
          <xs:sequence>
            <xs:element ref="SwitchAudit"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="IdentificationType">
      <xs:complexContent>
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

注 — 附录四介绍了SSL的使用。

12.3.4.5 MobileTwoFactorUnregistered

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregistered`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

该类型反映了没有移动用户注册的过程以及一个基于两要素的认证，例如安全设备和用户 PIN。该关联类型在除了移动运营商希望通过在注册时捕获移动电话数据将用户 ID 与移动提供的两要素认证业务联系起来之外的业务中有用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnre
gistered"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"

xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregistered"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">
```

```
    <xs:annotation>
      <xs:documentation>
       Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregistered
       Document identity:
saml-schema-authn-context-mobiletwofactor-unreg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="ZeroKnowledge"/>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
              <xs:element ref="AsymmetricDecryption"/>
              <xs:element ref="AsymmetricKeyAgreement"/>
              <xs:element ref="ComplexAuthenticator"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="ComplexAuthenticatorType">
      <xs:complexContent>
        <xs:restriction base="ComplexAuthenticatorType">
          <xs:sequence>
```

```
            <xs:choice>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
            </xs:choice>
            <xs:element ref="Password"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="MobileNetworkNoEncryption"/>
              <xs:element ref="MobileNetworkRadioEncryption"/>
              <xs:element ref="MobileNetworkEndToEndEncryption"/>
              <xs:element ref="WTLS"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="OperationalProtectionType">
      <xs:complexContent>
        <xs:restriction base="OperationalProtectionType">
          <xs:sequence>
            <xs:element ref="SecurityAudit"/>
            <xs:element ref="DeactivationCallCenter"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TechnicalProtectionBaseType">
      <xs:complexContent>
        <xs:restriction base="TechnicalProtectionBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PrivateKeyProtection"/>
              <xs:element ref="SecretKeyProtection"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrivateKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="PrivateKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecretKeyProtectionType">
      <xs:complexContent>
```

```
        <xs:restriction base="SecretKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="MobileDevice"/>
                <xs:enumeration value="MobileAuthCard"/>
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecurityAuditType">
      <xs:complexContent>
        <xs:restriction base="SecurityAuditType">
          <xs:sequence>
            <xs:element ref="SwitchAudit"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="IdentificationType">
      <xs:complexContent>
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.6 MobileOneFactorContract

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

该类型反映了移动用户注册过程和单要素认证。例如，一个具有抗干扰存储器（如：移动 MSISDN）存储密钥但没有要求的 PIN 或者用于实时用户认证的生物测定的数字信号设备。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema

targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract
"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract
        Document identity: saml-schema-authn-context-mobileonefactor-reg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
```

```
            <xs:element ref="ZeroKnowledge"/>
            <xs:element ref="SharedSecretChallengeResponse"/>
            <xs:element ref="SharedSecretDynamicPlaintext"/>
            <xs:element ref="AsymmetricDecryption"/>
            <xs:element ref="AsymmetricKeyAgreement"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="AuthenticatorTransportProtocolType">
    <xs:complexContent>
      <xs:restriction base="AuthenticatorTransportProtocolType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SSL"/>
            <xs:element ref="MobileNetworkNoEncryption"/>
            <xs:element ref="MobileNetworkRadioEncryption"/>
            <xs:element ref="MobileNetworkEndToEndEncryption"/>
            <xs:element ref="WTLS"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="OperationalProtectionType">
    <xs:complexContent>
      <xs:restriction base="OperationalProtectionType">
        <xs:sequence>
          <xs:element ref="SecurityAudit"/>
          <xs:element ref="DeactivationCallCenter"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="TechnicalProtectionBaseType">
    <xs:complexContent>
      <xs:restriction base="TechnicalProtectionBaseType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="PrivateKeyProtection"/>
            <xs:element ref="SecretKeyProtection"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="PrivateKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="PrivateKeyProtectionType">
        <xs:sequence>
          <xs:element ref="KeyStorage"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
```

```
    <xs:complexType name="SecretKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="SecretKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="smartcard"/>
                <xs:enumeration value="MobileDevice"/>
                <xs:enumeration value="MobileAuthCard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecurityAuditType">
      <xs:complexContent>
        <xs:restriction base="SecurityAuditType">
          <xs:sequence>
            <xs:element ref="SwitchAudit"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="IdentificationType">
      <xs:complexContent>
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="PhysicalVerification"/>
            <xs:element ref="WrittenConsent"/>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="verinymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.7 MobileTwoFactorContract

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

该类型反映了移动用户注册过程以及基于两要素的认证。例如，具有抗干扰存储器（如：GSM SIM）来存储密钥并需要清楚的证实用户标识和意向（如 PIN 或者生物测定）的数字信号设备。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorCont
ract"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract
        Document identity: saml-schema-authn-context-mobiletwofactor-reg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
```

```
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="ZeroKnowledge"/>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
              <xs:element ref="AsymmetricDecryption"/>
              <xs:element ref="AsymmetricKeyAgreement"/>
              <xs:element ref="ComplexAuthenticator"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="ComplexAuthenticatorType">
      <xs:complexContent>
        <xs:restriction base="ComplexAuthenticatorType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
            </xs:choice>
            <xs:element ref="Password"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="MobileNetworkNoEncryption"/>
              <xs:element ref="MobileNetworkRadioEncryption"/>
              <xs:element ref="MobileNetworkEndToEndEncryption"/>
              <xs:element ref="WTLS"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="OperationalProtectionType">
      <xs:complexContent>
        <xs:restriction base="OperationalProtectionType">
          <xs:sequence>
            <xs:element ref="SecurityAudit"/>
            <xs:element ref="DeactivationCallCenter"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TechnicalProtectionBaseType">
      <xs:complexContent>
        <xs:restriction base="TechnicalProtectionBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PrivateKeyProtection"/>
              <xs:element ref="SecretKeyProtection"/>
            </xs:choice>
```

```
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrivateKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="PrivateKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecretKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="SecretKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="MobileDevice"/>
                <xs:enumeration value="MobileAuthCard"/>
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecurityAuditType">
      <xs:complexContent>
        <xs:restriction base="SecurityAuditType">
          <xs:sequence>
            <xs:element ref="SwitchAudit"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="IdentificationType">
      <xs:complexContent>
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="PhysicalVerification"/>
            <xs:element ref="WrittenConsent"/>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
```

```
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="verinymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

注 — 附录四介绍了SSL的使用。

12.3.4.8 Password

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:Password`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

在主体通过不受保护的 HTTP 会话提交的密码向认证授权中心进行认证时密码类型是有用的。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Password
        Document identity: saml-schema-authn-context-pword-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
```

```
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

下面是符合该类型 Schema 的一个 XML 实例：

```
<AuthenticationContextDeclaration
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Password">

  <AuthnMethod>
    <Authenticator>
      <AuthenticatorSequence>
        <RestrictedPassword>
          <Length min="4"/>
        </RestrictedPassword>
      </AuthenticatorSequence>
    </Authenticator>
  </AuthnMethod>

</AuthenticationContextDeclaration>
```

12.3.4.9 PasswordProtectedTransport

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

在主体通过受保护的会话提交的密码向认证授权中心进行认证时 PasswordProtectedTransport 类型是有用的。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTra
nsport"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
        Document identity: saml-schema-authn-context-ppt-2.0
```

```
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="MobileNetworkRadioEncryption"/>
              <xs:element ref="MobileNetworkEndToEndEncryption"/>
              <xs:element ref="WTLS"/>
              <xs:element ref="IPSec"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

注 — 附录四介绍了SSL的使用。

12.3.4.10 PreviousSession

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:PreviousSession

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当责任人在之前的一些点使用认证授权中心支持的某些认证关联向认证授权中心认证过时，PreviousSession 类型是可用的。因此，认证授权中心随后将为信任方断言的认证事件可能在时间上被及时从主体当前的资源接入请求中明确的分开。

前面认证会话的关联很明显不包括在本关联类型中，因为用户在该会话期间并没有认证，并且因此用户在前面的会话中用来认证的机制不能被用做是否允许接入到资源的部分依据。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PreviousSession"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PreviousSession"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:PreviousSession
        Document identity: saml-schema-authn-context-session-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
```

```
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="PreviousSession"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.11 公开密钥 – X.509

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:X509

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

X509 关联类型表示责任人以数字签字的方式认证，并且它的密钥已证实作为 X.509 公开密钥基础设施的一部分。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:X509
        Document identity: saml-schema-authn-context-x509-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
```

```
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.12 公开密钥－PGP

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:PGP`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

PGP 关联类型表示责任人以数字签字的方式认证，并且它的密钥已证实作为 PGP 公开密钥基础设施的一部分有效的。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:PGP
        Document identity: saml-schema-authn-context-pgp-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
```

```
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.13 公开密钥－SPKI

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

SPKI 关联类型表示责任人以数字签字的方式认证，并且它的密钥已证实通过 SPKI 公开密钥基础设施的一部分有效的。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI
        Document identity: saml-schema-authn-context-spki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
```

```
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.14 公开密钥－XML数字签字

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此关联类型表示主体以数字签字的方式根据在 W3C XML 签字中规定的规则进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig
        Document identity: saml-schema-authn-context-xmldsig-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
           New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
```

```
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:ietf:rfc:3075"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.15 Smartcard

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当责任人使用智能卡向认证授权中心认证时，智能卡类型将被标识。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">
```

```
    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard
        Document identity: saml-schema-authn-context-smartcard-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="Smartcard"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.16 SmartcardPKI

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

在责任人使用带有私有密钥和 PIN 码的智能卡通过一个两要素认证机制向认证授权中心认证时 SmartcardPKI 类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI
        Document identity: saml-schema-authn-context-smartcardpki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TechnicalProtectionBaseType">
      <xs:complexContent>
        <xs:restriction base="TechnicalProtectionBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PrivateKeyProtection"/>
            </xs:choice>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
```

```
          <xs:sequence>
            <xs:element ref="Smartcard"/>
            <xs:element ref="ActivationPin"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="AsymmetricDecryption"/>
              <xs:element ref="AsymmetricKeyAgreement"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrivateKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="PrivateKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyActivationType">
      <xs:complexContent>
        <xs:restriction base="KeyActivationType">
          <xs:sequence>
            <xs:element ref="ActivationPin"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.17 SoftwarePKI

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当责任人使用存储在软件中的 X.509 证书向认证授权中心认证时，SoftwarePKI 类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI
        Document identity: saml-schema-authn-context-softwarepki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TechnicalProtectionBaseType">
      <xs:complexContent>
        <xs:restriction base="TechnicalProtectionBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PrivateKeyProtection"/>
```

```
            </xs:choice>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="ActivationPin"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="AsymmetricDecryption"/>
              <xs:element ref="AsymmetricKeyAgreement"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrivateKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="PrivateKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyActivationType">
      <xs:complexContent>
        <xs:restriction base="KeyActivationType">
          <xs:sequence>
            <xs:element ref="ActivationPin"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="memory"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
```

```
    </xs:complexType>

  </xs:redefine>
</xs:schema>
```

12.3.4.18 Telephony

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此类型用来表示责任人通过提供一个固定线路电话号码进行认证，该号码通过电话协议（例如 ADSL）传送。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony
        Document identity: saml-schema-authn-context-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
```

```
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SubscriberLineNumber"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.19 电话（游牧）

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此类型表示责任人正处于“漫游”中（可能使用一个电话卡）并以线路号码、用户前缀和密码元素的方式进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony
        Document identity: saml-schema-authn-context-nomad-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
```

```
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.20 电话（个性化的）

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalTelephony

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此类型用来表示责任人通过，例如 ADSL，传送一个固定线路电话号码和用户前缀的方式进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony"
```

```
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony
        Document identity: saml-schema-authn-context-personal-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
```

```
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.21 电话（已认证）

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此类型表示责任人通过线路号码、用户前缀和密码元素的方式进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTeleph
ony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony
        Document identity: saml-schema-authn-context-auth-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>
```

```
    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>
  </xs:redefine>

</xs:schema>
```

12.3.4.22 安全的远端密码

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当以 IETF RFC 2945 中规定的安全的远端密码的方式进行认证时，SecureRemotePassword 类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword
        Document identity: saml-schema-authn-context-srp-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
```

```
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SharedSecretChallengeResponse"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SharedSecretChallengeResponseType">
      <xs:complexContent>
        <xs:restriction base="SharedSecretChallengeResponseType">
          <xs:attribute name="method" type="xs:anyURI" fixed="urn:ietf:rfc:2945"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

12.3.4.23 基于TLS证书的客户端认证

URI: `urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient`

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

此类型表示责任人以一个使用 TLS 加密传送的客户端证书的方式进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">
```

```
  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient
        Document identity: saml-schema-authn-context-sslcert-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"/>
```

```
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="WTLS"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

注 — 附录四介绍了SSL的使用。

12.3.4.24 TimeSyncToken

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken

该 URI 也用做附件 A 中相应认证关联类别 Schema 的目标命名空间。

当责任人通过一个时间同步令牌进行认证时 TimeSyncToken 类型可用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken
        Document identity: saml-schema-authn-context-timesync-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
```

```
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"
minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="Token"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TokenType">
      <xs:complexContent>
        <xs:restriction base="TokenType">
          <xs:sequence>
            <xs:element ref="TimeSyncToken"/>
            <xs:element ref="Extension" minOccurs="0"
maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TimeSyncTokenType">
      <xs:complexContent>
        <xs:restriction base="TimeSyncTokenType">
          <xs:attribute name="DeviceType" use="required">
            <xs:simpleType>
              <xs:restriction base="DeviceTypeType">
                <xs:enumeration value="hardware"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>

          <xs:attribute name="SeedLength" use="required">
            <xs:simpleType>
              <xs:restriction base="xs:integer">
                <xs:minInclusive value="64"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>

          <xs:attribute name="DeviceInHand" use="required">
            <xs:simpleType>
              <xs:restriction base="booleanType">
                <xs:enumeration value="true"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>
```

```
  </xs:redefine>

</xs:schema>
```

12.3.4.25 未指定

URI: urn:oasis:names:tc:SAML:2.0:ac:classes:Unspecified

未指定的类型表示通过未指定的方式完成的认证。

13 SAML一致性要求

本部分描述了对 SAML 一致性断言的可选和必选的执行特征。

本建议书定义了大量已命名的协议子集，每个协议子集（除了属性协议子集）描述了选择 SAML 消息流的细节，而且可以被当做软件部分必须实现的功能。一个协议子集的实现包括在为每个消息交换使用的绑定。这样一个绑定可以视作完成一次消息交换的特定实现技巧。

本节列举了本建议书中定义的所有不同的协议子集。对于每个协议子集，列出了相关的 SAML V2.0 消息流，并且描述了每个消息流可能的一组映射。协议子集的结合、消息交换和一种选定的绑定被称作 SAML V2.0 的特征。

本节也描述了 SAML V2.0 的一致性矩阵。标识出一组不同的操作模式或角色。一致性矩阵描述了每个操作模式必须实现的特征集。

13.1 SAML协议子集和可能的执行

表 1 列举了 SAML 协议子集定义的所有的协议子集。对于每个协议子集，也描述了在协议子集中建立的消息协议流。对于每个消息流，在最后一列给出了一组相关的绑定。

表 1/X.1141－可能的实现

协议子集	消 息 流	绑 定
Web SSO	<AuthnRequest>从 SP 到 IdP	HTTP Redirect
		HTTP POST
		HTTP Artifact
	IdP <Response>到 SP	HTTP POST
		HTTP Artifact
Enhanced Client/Proxy SSO	ECP 到 SP, SP 到 ECP 到 IdP	PAOS
	IdP 到 ECP 到 SP, SP 到 ECP	PAOS
Identity Provider Discovery	Cookie setter	HTTP
	Cookie getter	HTTP
Single Logout	<LogoutRequest>	HTTP Redirect
		HTTP POST
		HTTP Artifact
		SOAP
	<LogoutResponse>	HTTP Redirect
		HTTP POST
		HTTP Artifact
		SOAP
Name Identifier Management	<ManageNameIDRequest>	HTTP Redirect
		HTTP POST
		HTTP Artifact
		SOAP
	<ManageNameIDResponse>	HTTP Redirect
		SOAP
Artifact Resolution	<ArtifactResolve>, <ArtifactResponse>	SOAP

表 1/X.1141－可能的实现

协议子集	消　息　流	绑　　定
Authentication Query	<AuthNQuery>, <Response>	SOAP
Attribute Query	<AttributeQuery>, <Response>	SOAP
Authorization Decision Query	<AuthZDecisionQuery>, <Response>	SOAP
采用标识符的断言请求	<AssertionIDRequest>, <Response>	SOAP
名称标识符映射	<NameIDMappingRequest>, <NameIDMappingResponse>	SOAP
SAML URI binding	GET, HTTP Response	HTTP
UUID 属性协议子集		
DCE PAC 属性协议子集		
X.500 属性协议子集		
XACML 属性协议子集		
元数据	使用	
	互换	

13.2　一致性

本节描述了 SAML V2.0 的技术一致性要求。

13.2.1　操作模式

本建议书中使用短语“操作模式”来描述软件构件在 SAML 一致性中所起的角色。共有如下操作模式：

- IdP－标识提供者
- IdP Lite－简单标识提供者
- SP－服务提供商
- SP Lite－简易服务提供商
- ECP－增强的客户端/代理
- SAML属性授权中心
- SAML决定授权的权威
- SAML认证授权中心
- SAML请求端

13.2.2 特征矩阵

下面的矩阵（见表 2）通过以从表 1 中获取的协议子集、消息、映射构成的三维形式标识出唯一的一致性要求集。当明显来自某个关联时，并不总是包括消息这一元素。

表 2/X.1141－特征矩阵

特　　征	IdP	IdP lite	SP	SP lite	ECP
Web SSO, <AuthnRequest>, HTTP redirect	MUST	MUST	MUST	MUST	N/A
Web SSO, <Response>, HTTP POST	MUST	MUST	MUST	MUST	N/A
Web SSO, <Response>, HTTP artifact	MUST	MUST	MUST	MUST	N/A
人为决定（Artifact Resolution）, SOAP	MUST	MUST	MUST	MUST	N/A
增强的客户端/代理 SSO, PAOS	MUST	MUST	MUST	MUST	MUST
命名标识符管理，HTTP 重定向（IdP-发起的）	MUST	MUST NOT	MUST	MUST NOT	N/A
命名标识符管理，SOAP（IdP-发起的）	MUST	MUST NOT	OPTIONAL	MUST NOT	N/A
名字标识管理，HTTP 重定向 注（资料性）－PE11（参见 OASIS PE:2006）建议添加（SP－发起的）	MUST	MUST NOT	MUST	MUST NOT	N/A
名字标识管理，SOAP(SP-发起的)	MUST	MUST NOT	OPTIONAL	MUST NOT	N/A
单点注销（IdP－发起的）－HTTP 重定向	MUST	MUST	MUST	MUST	N/A
单点注销（IdP－发起的）－SOAP	MUST	OPTIONAL	MUST	OPTIONAL	N/A
单点注销（SP－发起的）－HTTP 重定向	MUST	MUST	MUST	MUST	N/A
单点注销（SP－发起的）－SOAP	MUST	OPTIONAL	MUST	OPTIONAL	N/A
发现标识提供者（cookie）	MUST	MUST	OPTIONAL	OPTIONAL	N/A

注 1（资料性）— PE16（参见 OASIS PE:2006）建议在表 2 的最后一行最后一列以“N/A”代替“可选”。

注 2（资料性）— PE25（参见 OASIS PE:2006）建议在表 2 的后面增加如下内容：

特　　征	IdP	IdP Lite	SP	SP Lite	ECP
元数据结构	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL	N/A
元数据互操作	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL	N/A

注 3（资料性）— PE29（参见 OASIS PE:2006）建议在表 2 的后面增加如下内容：

特　　征	IdP	IdP Lite	SP	SP Lite	ECP
请求断言标识符	OPTIONAL	N/A	N/A	N/A	N/A
SAML URL 绑定	OPTIONAL	N/A	N/A	N/A	N/A

下面的表 3 总结出扩展了上面定义的 IdP 或 SP 模型的操作模式。这些将被理解为来自上表的 IdP 或 SP 模型与下面列出的相应的扩展特征的结合。

表 3/X.1141－扩展IdP和SP

特　　征	IdP扩展	SP扩展
标识提供者代理	MUST	MUST
名字标识者的映射，SOAP	MUST	MUST

下表总结了对 SAML 权威人士和请求端的一致性要求。

表 4/X.1141－SAML权威人士和请求端矩阵

特　　征	SAML认证 授权中心	SAML属性 授权中心	SAML决定 授权的权威	SAML请求端
认证查询，SOAP	MUST	OPTIONAL	OPTIONAL	OPTIONAL
属性查询，SOAP	OPTIONAL	MUST	OPTIONAL	OPTIONAL
决定授权的查询，SOAP	OPTIONAL	OPTIONAL	MUST	OPTIONAL
请求断言标识符，SOAP	MUST	MUST	MUST	OPTIONAL
SAML URI 绑定	MUST	MUST	MUST	OPTIONAL

注 4（资料性）— PE25 和 PE42（参见 OASIS PE:2006）建议将表 4 修改如下：

特　　征	SAML认证 授权中心	SAML属性 授权中心	SAML决定 授权的权威	SAML请求端
认证查询，SOAP	MUST	N/A	N/A	OPTIONAL
属性查询，SOAP	N/A	MUST	N/A	OPTIONAL
决定授权的查询，SOAP	N/A	N/A	MUST	OPTIONAL
请求断言标识符，SOAP	MUST	MUST	MUST	OPTIONAL
SAML URL 绑定	MUST	MUST	MUST	OPTIONAL
元数据结构	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL
元数据互操作	OPTIONAL	OPTIONAL	OPTIONAL	OPTIONAL

13.2.3 SAML所定义标识符的执行

所有相关的操作模式必须执行如下 SAML 定义的标识符：

- 第8节定义的All Attribute Name Format标识符
- 第8节定义的All Name Identifier Format标识符

当产生和使用 SAML 消息时，需确定 SAML 执行必须允许使用所有标识符常量（参见第 8.1 和 8.2 节）。SAML 消息发生器必须能够产生消息，SAML 消息使用者必须能够用在本节定义的任何常数处理消息。

第 8.17 节（永久命名的标识符）和第 10.1 节（临时命名的标识符）定义了这些标识符发生器的标准化处理规则。通过执行过程来证实这些小节中的所有标准化处理规则都必须被支持。剩余的其他标识符没有指定标准化处理规则。因此这些标识符的产生和使用仅在产生方和使用方在标识符的语义解释上达成外部定义的协议时才有意义。

注 — 在本关联中，“处理”意味着实施必须成功的解析和处理标识符，不能失败或返回错误信息。一旦标识符在这一水平处理，执行过程如何处理该标识符不在本建议书的范围内。

SAML 执行过程可以通过对标识符的直接执行支持或者使用支持的设计界面提供上面描述的工具。为了允许 SAML 执行过程必须能在设计上扩展以处理该执行过程未正常处理的所有标识符而提供这些界面。

13.2.4 加密元素的执行

在那些要求操作模式处理或产生相应的名字为 <saml:NameID>、<saml:Assertion> 或 <saml:Attribute>的未加密元素的任何关联中，所有相关的操作模式必须能够处理或产生下列加密元素：

- <saml:EncryptedID>
- <saml:EncryptedAssertion>
- <saml:EncryptedAttribute>

13.2.5 SOAP和URL绑定安全模型

下列安全模型对所有使用 SOAP 绑定执行的所有协议子集以及 SAML URL 绑定都是必备执行的。SAML 授权中心和请求端必须执行下列认证方法：

- 无客户端和服务器认证。
- 使用或不使用TLS 1.0的HTTP基础认证，SAML请求端必须抢先发送带有初始请求的许可头。
- 基于TLS 1.0服务器的HTTP使用服务器证书进行认证。
- 基于TLS 1.0的HTTP使用服务器和客户端两种证书进行相互认证。

如果一个 SAML 权威使用 TLS 1.0，就必须使用服务器证书。

注 1（资料性）— PE25（参见OASIS PE:2006）建议新增加一个有关元数据结构的小节，如下：

SAML的执行断言一致性可能通过选出元数据结构选项来宣布SAML元数据每个操作模式的一致性。至于每个操作模式，此类一致性的细节如下：

根据在互操作节点具有依赖于SAML元数据存在的选项（如SAML规范中的规定）的所有情况中可扩展的SAML元数据格式执行SAML元数据。选择元数据结构选项具有要求这些元数据对互操作节点可用的效果。如下所述，在元数据互操作特征中提供一种满足该要求的方式。

因此，当与互操作节点、特定操作以及当前交换有关的已知元数据已过期或者在缓存中已失效时，对互操作节点SAML元数据的参考、使用和忠诚保证该元数据可用并且不被政策或特定操作以及该特定交换禁止。

注 2（资料性）—PE25（参见OASIS PE:2006）建议新增加一个有关元数据互操作的小节，如下：

元数据互操作选项的选择要求执行过程提供除了任何其他机制之外的在第9节中SAML元数据中知名的位置公布和决议机制。

13.3 XML数字签字和XML加密

SAML V2.0 为完整性，使用 XML 签字来执行 XML 标记和加密功能以及源认证。SAML V2.0 使用 XML 加密来保证机密性，包括加密标识符、加密断言和加密属性。

13.3.1 XML签字算法XML

第 6.1 节规定的 W3C XML 签字，强制使用下列算法：

- 摘要：SHA-1；
- MAC：HMAC-SHA1；
- XML 规范：CanonicalXML（无注解）；
- 转换：包封的签字。

因此他们必须由顺应 SAML V2.0 执行过程执行。

此外，为能够互操作，顺应 SAML V2.0 执行过程必须执行下面的算法：

- 签字：RSA和SHA1（W3C签字中推荐使用，互操作需要该算法）。

尽管 XML 签字必须使用 DSA 和 SHA1 签字算法，SAML V2.0 则没有强制要求该算法，但建议使用。

注 — 现在NIST（国家标准和技术协会）目前鼓励使用SHA-256（使用256比特密钥的安全哈希算法）代替SHA-1。

13.3.2 XML加密算法

- W3C XML加密，第5.2.1和5.2.2节强制使用下列算法：块加密：三倍DES、AES-128、AES-256。
- 密钥传送：RSA-v1.5、RSA-OAEP。

因此上述算法必须由遵循 SAML V2.0 执行过程执行。

13.4 TLS 1.0的使用

在任何使用 TLS 1.0 的 SAML V2.0 中，服务器必须使用 X.509 v3 证书来认证客户端。客户端必须根据证书的内容建立服务器身份（典型的做法是通过检查证书的主体 DN 字段）。

13.4.1 SAML SOAP和URL绑定

具有 TLS 能力的执行过程必须采用 TLS_RSA_WITH_3DES_EDE_CBC_SHA 密码组，而且有可能采用 TLS_RSA_AES_128_CBC_SHA 密码组。

就有 FIPS TLS 能力的执行过程必须采用相应的 TLS_RSA_FIPS_WITH_3DES_EDE_CBC_SHA 密码组，而且有可能采用 TLS_RSA_FIPS_AES_128_CBC_SHA 密码组。

13.4.2 SAML的Web SSO协议子集

具有 TLS 能力的执行过程必须采用 TLS_RSA_WITH_3DES_EDE_CBC_SHA 密码组(参见 IETF RFC 2246)。

附 件 A

SAML Schema

A.1 SAML Schema断言

SAML Schema 断言如下：

```
<?xml version="1.0" encoding="US-ASCII"?>
<schema
    targetNamespace="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="http://www.w3.org/2000/09/xmldsig#"

schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/xmldsig-core
-schema.xsd"/>
    <import namespace="http://www.w3.org/2001/04/xmlenc#"

schemaLocation="http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/xenc-schema.x
sd"/>
    <annotation>
        <documentation>
            Document identifier: saml-schema-assertion-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V1.0 (November, 2002):
              Initial Standard Schema.
            V1.1 (September, 2003):
              Updates within the same V1.0 namespace.
            V2.0 (March, 2005):
              New assertion schema for SAML V2.0 namespace.
        </documentation>
    </annotation>
    <attributeGroup name="IDNameQualifiers">
        <attribute name="NameQualifier" type="string" use="optional"/>
        <attribute name="SPNameQualifier" type="string" use="optional"/>
    </attributeGroup>
    <element name="BaseID" type="saml:BaseIDAbstractType"/>
    <complexType name="BaseIDAbstractType" abstract="true">
        <attributeGroup ref="saml:IDNameQualifiers"/>
    </complexType>
    <element name="NameID" type="saml:NameIDType"/>
    <complexType name="NameIDType">
        <simpleContent>
```

```
            <extension base="string">
                <attributeGroup ref="saml:IDNameQualifiers"/>
                <attribute name="Format" type="anyURI" use="optional"/>
                <attribute name="SPProvidedID" type="string" use="optional"/>
            </extension>
        </simpleContent>
    </complexType>
    <complexType name="EncryptedElementType">
        <sequence>
            <element ref="xenc:EncryptedData"/>
            <element ref="xenc:EncryptedKey" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <element name="EncryptedID" type="saml:EncryptedElementType"/>
    <element name="Issuer" type="saml:NameIDType"/>
    <element name="AssertionIDRef" type="NCName"/>
    <element name="AssertionURIRef" type="anyURI"/>
    <element name="Assertion" type="saml:AssertionType"/>
    <complexType name="AssertionType">
        <sequence>
            <element ref="saml:Issuer"/>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="saml:Subject" minOccurs="0"/>
            <element ref="saml:Conditions" minOccurs="0"/>
            <element ref="saml:Advice" minOccurs="0"/>
            <choice minOccurs="0" maxOccurs="unbounded">
                <element ref="saml:Statement"/>
                <element ref="saml:AuthnStatement"/>
                <element ref="saml:AuthzDecisionStatement"/>
                <element ref="saml:AttributeStatement"/>
            </choice>
        </sequence>
        <attribute name="Version" type="string" use="required"/>
        <attribute name="ID" type="ID" use="required"/>
        <attribute name="IssueInstant" type="dateTime" use="required"/>
    </complexType>
    <element name="Subject" type="saml:SubjectType"/>
    <complexType name="SubjectType">
        <choice>
            <sequence>
                <choice>
                    <element ref="saml:BaseID"/>
                    <element ref="saml:NameID"/>
                    <element ref="saml:EncryptedID"/>
                </choice>
                <element ref="saml:SubjectConfirmation" minOccurs="0"
maxOccurs="unbounded"/>
            </sequence>
            <element ref="saml:SubjectConfirmation" maxOccurs="unbounded"/>
        </choice>
    </complexType>
    <element name="SubjectConfirmation" type="saml:SubjectConfirmationType"/>
    <complexType name="SubjectConfirmationType">
        <sequence>
            <choice minOccurs="0">
                <element ref="saml:BaseID"/>
                <element ref="saml:NameID"/>
                <element ref="saml:EncryptedID"/>
            </choice>
            <element ref="saml:SubjectConfirmationData" minOccurs="0"/>
        </sequence>
        <attribute name="Method" type="anyURI" use="required"/>
    </complexType>
    <element name="SubjectConfirmationData"
type="saml:SubjectConfirmationDataType"/>
    <complexType name="SubjectConfirmationDataType" mixed="true">
        <complexContent>
            <restriction base="anyType">
                <sequence>
                    <any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
```

```
            </sequence>
            <attribute name="NotBefore" type="dateTime" use="optional"/>
            <attribute name="NotOnOrAfter" type="dateTime" use="optional"/>
            <attribute name="Recipient" type="anyURI" use="optional"/>
            <attribute name="InResponseTo" type="NCName" use="optional"/>
            <attribute name="Address" type="string" use="optional"/>
            <anyAttribute namespace="##other" processContents="lax"/>
        </restriction>
    </complexContent>
</complexType>
<complexType name="KeyInfoConfirmationDataType" mixed="false">
    <complexContent>
        <restriction base="saml:SubjectConfirmationDataType">
            <sequence>
                <element ref="ds:KeyInfo" maxOccurs="unbounded"/>
            </sequence>
        </restriction>
    </complexContent>
</complexType>
<element name="Conditions" type="saml:ConditionsType"/>
<complexType name="ConditionsType">
    <choice minOccurs="0" maxOccurs="unbounded">
        <element ref="saml:Condition"/>
        <element ref="saml:AudienceRestriction"/>
        <element ref="saml:OneTimeUse"/>
        <element ref="saml:ProxyRestriction"/>
    </choice>
    <attribute name="NotBefore" type="dateTime" use="optional"/>
    <attribute name="NotOnOrAfter" type="dateTime" use="optional"/>
</complexType>
<element name="Condition" type="saml:ConditionAbstractType"/>
<complexType name="ConditionAbstractType" abstract="true"/>
<element name="AudienceRestriction" type="saml:AudienceRestrictionType"/>
<complexType name="AudienceRestrictionType">
    <complexContent>
        <extension base="saml:ConditionAbstractType">
            <sequence>
                <element ref="saml:Audience" maxOccurs="unbounded"/>
            </sequence>
        </extension>
    </complexContent>
</complexType>
<element name="Audience" type="anyURI"/>
<element name="OneTimeUse" type="saml:OneTimeUseType" />
<complexType name="OneTimeUseType">
    <complexContent>
        <extension base="saml:ConditionAbstractType"/>
    </complexContent>
</complexType>
<element name="ProxyRestriction" type="saml:ProxyRestrictionType"/>
<complexType name="ProxyRestrictionType">
<complexContent>
    <extension base="saml:ConditionAbstractType">
        <sequence>
            <element ref="saml:Audience" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
        <attribute name="Count" type="nonNegativeInteger" use="optional"/>
    </extension>
</complexContent>
</complexType>
<element name="Advice" type="saml:AdviceType"/>
<complexType name="AdviceType">
    <choice minOccurs="0" maxOccurs="unbounded">
        <element ref="saml:AssertionIDRef"/>
        <element ref="saml:AssertionURIRef"/>
        <element ref="saml:Assertion"/>
        <element ref="saml:EncryptedAssertion"/>
        <any namespace="##other" processContents="lax"/>
    </choice>
</complexType>
<element name="EncryptedAssertion" type="saml:EncryptedElementType"/>
```

```
    <element name="Statement" type="saml:StatementAbstractType"/>
    <complexType name="StatementAbstractType" abstract="true"/>
    <element name="AuthnStatement" type="saml:AuthnStatementType"/>
    <complexType name="AuthnStatementType">
        <complexContent>
            <extension base="saml:StatementAbstractType">
                <sequence>
                    <element ref="saml:SubjectLocality" minOccurs="0"/>
                    <element ref="saml:AuthnContext"/>
                </sequence>
                <attribute name="AuthnInstant" type="dateTime" use="required"/>
                <attribute name="SessionIndex" type="string" use="optional"/>
                <attribute name="SessionNotOnOrAfter" type="dateTime"
use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="SubjectLocality" type="saml:SubjectLocalityType"/>
    <complexType name="SubjectLocalityType">
        <attribute name="Address" type="string" use="optional"/>
        <attribute name="DNSName" type="string" use="optional"/>
    </complexType>
    <element name="AuthnContext" type="saml:AuthnContextType"/>
    <complexType name="AuthnContextType">
        <sequence>
            <choice>
                <sequence>
                    <element ref="saml:AuthnContextClassRef"/>
                    <choice minOccurs="0">
                        <element ref="saml:AuthnContextDecl"/>
                        <element ref="saml:AuthnContextDeclRef"/>
                    </choice>
                </sequence>
                <choice>
                    <element ref="saml:AuthnContextDecl"/>
                    <element ref="saml:AuthnContextDeclRef"/>
                </choice>
            </choice>
            <element ref="saml:AuthenticatingAuthority" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <element name="AuthnContextClassRef" type="anyURI"/>
    <element name="AuthnContextDeclRef" type="anyURI"/>
    <element name="AuthnContextDecl" type="anyType"/>
    <element name="AuthenticatingAuthority" type="anyURI"/>
    <element name="AuthzDecisionStatement"
type="saml:AuthzDecisionStatementType"/>
    <complexType name="AuthzDecisionStatementType">
        <complexContent>
            <extension base="saml:StatementAbstractType">
                <sequence>
                    <element ref="saml:Action" maxOccurs="unbounded"/>
                    <element ref="saml:Evidence" minOccurs="0"/>
                </sequence>
                <attribute name="Resource" type="anyURI" use="required"/>
                <attribute name="Decision" type="saml:DecisionType"
use="required"/>
            </extension>
        </complexContent>
    </complexType>
    <simpleType name="DecisionType">
        <restriction base="string">
            <enumeration value="Permit"/>
            <enumeration value="Deny"/>
            <enumeration value="Indeterminate"/>
        </restriction>
    </simpleType>
    <element name="Action" type="saml:ActionType"/>
    <complexType name="ActionType">
        <simpleContent>
```

```
            <extension base="string">
                <attribute name="Namespace" type="anyURI" use="required"/>
            </extension>
        </simpleContent>
    </complexType>
    <element name="Evidence" type="saml:EvidenceType"/>
    <complexType name="EvidenceType">
        <choice maxOccurs="unbounded">
            <element ref="saml:AssertionIDRef"/>
            <element ref="saml:AssertionURIRef"/>
            <element ref="saml:Assertion"/>
            <element ref="saml:EncryptedAssertion"/>
        </choice>
    </complexType>
    <element name="AttributeStatement" type="saml:AttributeStatementType"/>
    <complexType name="AttributeStatementType">
        <complexContent>
            <extension base="saml:StatementAbstractType">
                <choice maxOccurs="unbounded">
                    <element ref="saml:Attribute"/>
                    <element ref="saml:EncryptedAttribute"/>
                </choice>
            </extension>
        </complexContent>
    </complexType>
    <element name="Attribute" type="saml:AttributeType"/>
    <complexType name="AttributeType">
        <sequence>
            <element ref="saml:AttributeValue" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
        <attribute name="Name" type="string" use="required"/>
        <attribute name="NameFormat" type="anyURI" use="optional"/>
        <attribute name="FriendlyName" type="string" use="optional"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>
    <element name="AttributeValue" type="anyType" nillable="true"/>
    <element name="EncryptedAttribute" type="saml:EncryptedElementType"/>
</schema>
```

A.2 SAML Schema认证关联

SAML 认证关联 Schema：

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="urn:oasis:names:tc:SAML:2.0:ac"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac"
  blockDefault="substitution"
  version="2.0">

  <xs:annotation>
    <xs:documentation>
      Document identifier: saml-schema-authn-context-2.0
      Location: http://docs.oasis-open.org/security/saml/v2.0/
      Revision history:
        V2.0 (March, 2005):
          New core authentication context schema for SAML V2.0.
          This is just an include of all types from the Shema
          referred to in the include statement below.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="saml-schema-authn-context-types-2.0.xsd"/>

</xs:schema>
```

A.3 SAML Schema认证关联 — AuthenticatedTelephony

下面是与 SAML 认证关联 Schema 相关的电话。

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:AuthenticatedTelephony
        Document identifier: saml-schema-authn-context-auth-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
```

```
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.4 SAML Schema认证关联 — IP

下表列出指定 IP 的 SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
  targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocol
        Document identifier: saml-schema-authn-context-ip-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
```

```
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="IPAddress"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.5 SAML Schema认证关联 — IPPWord

下表列出网际协议密钥（IPPword）SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword
"
  xmlns:ac="urn:oasis:names:tc:SAML:2.0:ac"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:InternetProtocolPassword
        Document identifier: saml-schema-authn-context-ippword-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
```

```
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="IPAddress"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.6 SAML Schema认证关联 — Kerberos

下表列出 SAML Kerberos 认证 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos
        Document identifier: saml-schema-authn-context-kerberos-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
```

```
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SharedSecretChallengeResponse"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SharedSecretChallengeResponseType">
      <xs:complexContent>
        <xs:restriction base="SharedSecretChallengeResponseType">
          <xs:attribute name="method" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>
</xs:schema>
```

A.7 SAML Schema认证关联 — MobileOneFactor-reg

下表包含用语注册的 MobileOneFactorContract SAML 关联文类别 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorContract
        Document identifier: saml-schema-authn-context-mobileonefactor-reg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>
```

```
<xs:complexType name="AuthnContextDeclarationBaseType">
  <xs:complexContent>
    <xs:restriction base="AuthnContextDeclarationBaseType">
      <xs:sequence>
        <xs:element ref="Identification" minOccurs="0"/>
        <xs:element ref="TechnicalProtection" minOccurs="0"/>
        <xs:element ref="OperationalProtection" minOccurs="0"/>
        <xs:element ref="AuthnMethod"/>
        <xs:element ref="GoverningAgreements" minOccurs="0"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="ID" type="xs:ID" use="optional"/>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="AuthnMethodBaseType">
  <xs:complexContent>
    <xs:restriction base="AuthnMethodBaseType">
      <xs:sequence>
        <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
        <xs:element ref="Authenticator"/>
        <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="AuthenticatorBaseType">
  <xs:complexContent>
    <xs:restriction base="AuthenticatorBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="DigSig"/>
          <xs:element ref="ZeroKnowledge"/>
          <xs:element ref="SharedSecretChallengeResponse"/>
          <xs:element ref="SharedSecretDynamicPlaintext"/>
          <xs:element ref="AsymmetricDecryption"/>
          <xs:element ref="AsymmetricKeyAgreement"/>
        </xs:choice>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="AuthenticatorTransportProtocolType">
  <xs:complexContent>
    <xs:restriction base="AuthenticatorTransportProtocolType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="SSL"/>
          <xs:element ref="MobileNetworkNoEncryption"/>
          <xs:element ref="MobileNetworkRadioEncryption"/>
          <xs:element ref="MobileNetworkEndToEndEncryption"/>
          <xs:element ref="WTLS"/>
        </xs:choice>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="OperationalProtectionType">
  <xs:complexContent>
    <xs:restriction base="OperationalProtectionType">
      <xs:sequence>
        <xs:element ref="SecurityAudit"/>
        <xs:element ref="DeactivationCallCenter"/>
```

```
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="TechnicalProtectionBaseType">
  <xs:complexContent>
    <xs:restriction base="TechnicalProtectionBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="PrivateKeyProtection"/>
          <xs:element ref="SecretKeyProtection"/>
        </xs:choice>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PrivateKeyProtectionType">
  <xs:complexContent>
    <xs:restriction base="PrivateKeyProtectionType">
      <xs:sequence>
        <xs:element ref="KeyStorage"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="SecretKeyProtectionType">
  <xs:complexContent>
    <xs:restriction base="SecretKeyProtectionType">
      <xs:sequence>
        <xs:element ref="KeyStorage"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="KeyStorageType">
  <xs:complexContent>
    <xs:restriction base="KeyStorageType">
      <xs:attribute name="medium" use="required">
        <xs:simpleType>
          <xs:restriction base="mediumType">
            <xs:enumeration value="smartcard"/>
            <xs:enumeration value="MobileDevice"/>
            <xs:enumeration value="MobileAuthCard"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="SecurityAuditType">
  <xs:complexContent>
    <xs:restriction base="SecurityAuditType">
      <xs:sequence>
        <xs:element ref="SwitchAudit"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="IdentificationType">
  <xs:complexContent>
```

```
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="PhysicalVerification"/>
            <xs:element ref="WrittenConsent"/>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="verinymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.8 SAML Schema认证关联 — MobileOneFactor-unreg

下表包含用于未注册的 MobileOneFactorContract SAML 关联类别 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregiste
red"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregistered"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileOneFactorUnregistered
        Document identifier: saml-schema-authn-context-mobileonefactor-unreg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>
```

```
  <xs:complexType name="AuthnMethodBaseType">
    <xs:complexContent>
      <xs:restriction base="AuthnMethodBaseType">
        <xs:sequence>
          <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
          <xs:element ref="Authenticator"/>
          <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="AuthenticatorBaseType">
    <xs:complexContent>
      <xs:restriction base="AuthenticatorBaseType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="DigSig"/>
            <xs:element ref="ZeroKnowledge"/>
            <xs:element ref="SharedSecretChallengeResponse"/>
            <xs:element ref="SharedSecretDynamicPlaintext"/>
            <xs:element ref="AsymmetricDecryption"/>
            <xs:element ref="AsymmetricKeyAgreement"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="AuthenticatorTransportProtocolType">
    <xs:complexContent>
      <xs:restriction base="AuthenticatorTransportProtocolType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SSL"/>
            <xs:element ref="MobileNetworkNoEncryption"/>
            <xs:element ref="MobileNetworkRadioEncryption"/>
            <xs:element ref="MobileNetworkEndToEndEncryption"/>
            <xs:element ref="WTLS"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="OperationalProtectionType">
    <xs:complexContent>
      <xs:restriction base="OperationalProtectionType">
        <xs:sequence>
          <xs:element ref="SecurityAudit"/>
          <xs:element ref="DeactivationCallCenter"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="TechnicalProtectionBaseType">
    <xs:complexContent>
      <xs:restriction base="TechnicalProtectionBaseType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="PrivateKeyProtection"/>
            <xs:element ref="SecretKeyProtection"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
```

```
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="PrivateKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="PrivateKeyProtectionType">
        <xs:sequence>
          <xs:element ref="KeyStorage"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="SecretKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="SecretKeyProtectionType">
        <xs:sequence>
          <xs:element ref="KeyStorage"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="KeyStorageType">
    <xs:complexContent>
      <xs:restriction base="KeyStorageType">
        <xs:attribute name="medium" use="required">
          <xs:simpleType>
            <xs:restriction base="mediumType">
              <xs:enumeration value="MobileDevice"/>
              <xs:enumeration value="MobileAuthCard"/>
              <xs:enumeration value="smartcard"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="SecurityAuditType">
    <xs:complexContent>
      <xs:restriction base="SecurityAuditType">
        <xs:sequence>
          <xs:element ref="SwitchAudit"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="IdentificationType">
    <xs:complexContent>
      <xs:restriction base="IdentificationType">
        <xs:sequence>
          <xs:element ref="GoverningAgreements"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="nym">
          <xs:simpleType>
            <xs:restriction base="nymType">
              <xs:enumeration value="anonymity"/>
              <xs:enumeration value="pseudonymity"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
```

```
  </xs:redefine>

</xs:schema>
```

A.9 SAML Schema认证关联 — MobileTwoFactor-reg

下表包含用于注册的 MobileTwoFactorContracSAML 关联类别 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorContract
        Document identifier: saml-schema-authn-context-mobiletwofactor-reg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="ZeroKnowledge"/>
              <xs:element ref="SharedSecretChallengeResponse"/>
```

```
            <xs:element ref="SharedSecretDynamicPlaintext"/>
            <xs:element ref="AsymmetricDecryption"/>
            <xs:element ref="AsymmetricKeyAgreement"/>
            <xs:element ref="ComplexAuthenticator"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="ComplexAuthenticatorType">
    <xs:complexContent>
      <xs:restriction base="ComplexAuthenticatorType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SharedSecretChallengeResponse"/>
            <xs:element ref="SharedSecretDynamicPlaintext"/>
          </xs:choice>
          <xs:element ref="Password"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="AuthenticatorTransportProtocolType">
    <xs:complexContent>
      <xs:restriction base="AuthenticatorTransportProtocolType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SSL"/>
            <xs:element ref="MobileNetworkNoEncryption"/>
            <xs:element ref="MobileNetworkRadioEncryption"/>
            <xs:element ref="MobileNetworkEndToEndEncryption"/>
            <xs:element ref="WTLS"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="OperationalProtectionType">
    <xs:complexContent>
      <xs:restriction base="OperationalProtectionType">
        <xs:sequence>
          <xs:element ref="SecurityAudit"/>
          <xs:element ref="DeactivationCallCenter"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="TechnicalProtectionBaseType">
    <xs:complexContent>
      <xs:restriction base="TechnicalProtectionBaseType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="PrivateKeyProtection"/>
            <xs:element ref="SecretKeyProtection"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="PrivateKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="PrivateKeyProtectionType">
```

```
        <xs:sequence>
          <xs:element ref="KeyActivation"/>
          <xs:element ref="KeyStorage"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="SecretKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="SecretKeyProtectionType">
        <xs:sequence>
          <xs:element ref="KeyActivation"/>
          <xs:element ref="KeyStorage"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="KeyStorageType">
    <xs:complexContent>
      <xs:restriction base="KeyStorageType">
        <xs:attribute name="medium" use="required">
          <xs:simpleType>
            <xs:restriction base="mediumType">
              <xs:enumeration value="MobileDevice"/>
              <xs:enumeration value="MobileAuthCard"/>
              <xs:enumeration value="smartcard"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="SecurityAuditType">
    <xs:complexContent>
      <xs:restriction base="SecurityAuditType">
        <xs:sequence>
          <xs:element ref="SwitchAudit"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="IdentificationType">
    <xs:complexContent>
      <xs:restriction base="IdentificationType">
        <xs:sequence>
          <xs:element ref="PhysicalVerification"/>
          <xs:element ref="WrittenConsent"/>
          <xs:element ref="GoverningAgreements"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="nym">
          <xs:simpleType>
            <xs:restriction base="nymType">
              <xs:enumeration value="anonymity"/>
              <xs:enumeration value="verinymity"/>
              <xs:enumeration value="pseudonymity"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:redefine>
```

```
</xs:schema>
```

A.10 SAML Schema认证关联 — MobileTwoFactor-unreg

下表包含基于 MobileTwoFactorUnregistered SAML 关联类别 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregiste
red"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregistered"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:MobileTwoFactorUnregistered
        Document identifier: saml-schema-authn-context-mobiletwofactor-unreg-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="DigSig"/>
              <xs:element ref="ZeroKnowledge"/>
              <xs:element ref="SharedSecretChallengeResponse"/>
              <xs:element ref="SharedSecretDynamicPlaintext"/>
```

```
            <xs:element ref="AsymmetricDecryption"/>
            <xs:element ref="AsymmetricKeyAgreement"/>
            <xs:element ref="ComplexAuthenticator"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="ComplexAuthenticatorType">
    <xs:complexContent>
      <xs:restriction base="ComplexAuthenticatorType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SharedSecretChallengeResponse"/>
            <xs:element ref="SharedSecretDynamicPlaintext"/>
          </xs:choice>
          <xs:element ref="Password"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="AuthenticatorTransportProtocolType">
    <xs:complexContent>
      <xs:restriction base="AuthenticatorTransportProtocolType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="SSL"/>
            <xs:element ref="MobileNetworkNoEncryption"/>
            <xs:element ref="MobileNetworkRadioEncryption"/>
            <xs:element ref="MobileNetworkEndToEndEncryption"/>
            <xs:element ref="WTLS"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="OperationalProtectionType">
    <xs:complexContent>
      <xs:restriction base="OperationalProtectionType">
        <xs:sequence>
          <xs:element ref="SecurityAudit"/>
          <xs:element ref="DeactivationCallCenter"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="TechnicalProtectionBaseType">
    <xs:complexContent>
      <xs:restriction base="TechnicalProtectionBaseType">
        <xs:sequence>
          <xs:choice>
            <xs:element ref="PrivateKeyProtection"/>
            <xs:element ref="SecretKeyProtection"/>
          </xs:choice>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="PrivateKeyProtectionType">
    <xs:complexContent>
      <xs:restriction base="PrivateKeyProtectionType">
        <xs:sequence>
```

```
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecretKeyProtectionType">
      <xs:complexContent>
        <xs:restriction base="SecretKeyProtectionType">
          <xs:sequence>
            <xs:element ref="KeyActivation"/>
            <xs:element ref="KeyStorage"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="KeyStorageType">
      <xs:complexContent>
        <xs:restriction base="KeyStorageType">
          <xs:attribute name="medium" use="required">
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="MobileDevice"/>
                <xs:enumeration value="MobileAuthCard"/>
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SecurityAuditType">
      <xs:complexContent>
        <xs:restriction base="SecurityAuditType">
          <xs:sequence>
            <xs:element ref="SwitchAudit"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="IdentificationType">
      <xs:complexContent>
        <xs:restriction base="IdentificationType">
          <xs:sequence>
            <xs:element ref="GoverningAgreements"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="nym">
            <xs:simpleType>
              <xs:restriction base="nymType">
                <xs:enumeration value="anonymity"/>
                <xs:enumeration value="pseudonymity"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.11 SAML Schema认证关联 — NomadTelephony

下表包含 SAML 游牧电话认证 Schema。游牧电话的责任人是当“漫游”时通过线路号码、用户后缀、密码元素等方法进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:NomadTelephony
        Document identifier: saml-schema-authn-context-nomad-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="Password"/>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
```

```
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.12 SAML Schema认证关联 — PersonalizedTelephony

下表提供用于个人电话的 SAML 认证 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:PersonalizedTelephony
        Document identifier: saml-schema-authn-context-personal-telephony-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
```

```
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SubscriberLineNumber"/>
            <xs:element ref="UserSuffix"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.13 SAML Schema认证关联 — PGP

下表提供基于 PGP 的 SAML 认证 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:PGP
        Document identifier: saml-schema-authn-context-pgp-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
```

```
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:PGP"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.14 SAML Schema认证关联 — PPT

下表包含基于密码保护的传送 SAML 认证 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTranspo
rt"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
        Document identifier: saml-schema-authn-context-ppt-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
```

```
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="MobileNetworkRadioEncryption"/>
              <xs:element ref="MobileNetworkEndToEndEncryption"/>
              <xs:element ref="WTLS"/>
              <xs:element ref="IPSec"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.15 SAML Schema认证关联 — Password

下表包含 SAML 认证关联密码 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Password
        Document identifier: saml-schema-authn-context-pword-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
```

```
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.16 SAML Schema认证关联 — PrevieusSession

下表包含基于前一次会话的SAML认证关联Schema。当责任人已经在过去的某点向认证授权中心进行认证，使用认证授权中心支持的认证关联。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Password"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Password
        Document identifier: saml-schema-authn-context-pword-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
```

```
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.17 SAML Schema认证关联 — Smartcard

下表列出智能卡 SAML 认证关联 Schema：

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Smartcard
        Document identifier: saml-schema-authn-context-smartcard-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
```

```
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="Smartcard"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.18 SAML Schema认证关联 — SmartcardPKI

下表列出 PKI 智能卡 SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI
        Document identifier: saml-schema-authn-context-smartcardpki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
```

```
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="TechnicalProtectionBaseType">
  <xs:complexContent>
    <xs:restriction base="TechnicalProtectionBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="PrivateKeyProtection"/>
        </xs:choice>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PrincipalAuthenticationMechanismType">
  <xs:complexContent>
    <xs:restriction base="PrincipalAuthenticationMechanismType">
      <xs:sequence>
        <xs:element ref="Smartcard"/>
        <xs:element ref="ActivationPin"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="AuthenticatorBaseType">
  <xs:complexContent>
    <xs:restriction base="AuthenticatorBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="DigSig"/>
          <xs:element ref="AsymmetricDecryption"/>
          <xs:element ref="AsymmetricKeyAgreement"/>
        </xs:choice>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PrivateKeyProtectionType">
  <xs:complexContent>
    <xs:restriction base="PrivateKeyProtectionType">
      <xs:sequence>
        <xs:element ref="KeyActivation"/>
        <xs:element ref="KeyStorage"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="KeyActivationType">
  <xs:complexContent>
    <xs:restriction base="KeyActivationType">
      <xs:sequence>
        <xs:element ref="ActivationPin"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="KeyStorageType">
  <xs:complexContent>
    <xs:restriction base="KeyStorageType">
      <xs:attribute name="medium" use="required">
```

```
            <xs:simpleType>
              <xs:restriction base="mediumType">
                <xs:enumeration value="smartcard"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.19 SAML Schema认证关联 — SoftwarePKI

下表列出软件 PKI SMAL 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SoftwarePKI
        Document identifier: saml-schema-authn-context-softwarepki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>
```

```
<xs:complexType name="TechnicalProtectionBaseType">
  <xs:complexContent>
    <xs:restriction base="TechnicalProtectionBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="PrivateKeyProtection"/>
        </xs:choice>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PrincipalAuthenticationMechanismType">
  <xs:complexContent>
    <xs:restriction base="PrincipalAuthenticationMechanismType">
      <xs:sequence>
        <xs:element ref="ActivationPin"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="AuthenticatorBaseType">
  <xs:complexContent>
    <xs:restriction base="AuthenticatorBaseType">
      <xs:sequence>
        <xs:choice>
          <xs:element ref="DigSig"/>
          <xs:element ref="AsymmetricDecryption"/>
          <xs:element ref="AsymmetricKeyAgreement"/>
        </xs:choice>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PrivateKeyProtectionType">
  <xs:complexContent>
    <xs:restriction base="PrivateKeyProtectionType">
      <xs:sequence>
        <xs:element ref="KeyActivation"/>
        <xs:element ref="KeyStorage"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="KeyActivationType">
  <xs:complexContent>
    <xs:restriction base="KeyActivationType">
      <xs:sequence>
        <xs:element ref="ActivationPin"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="KeyStorageType">
  <xs:complexContent>
    <xs:restriction base="KeyStorageType">
      <xs:attribute name="medium" use="required">
        <xs:simpleType>
          <xs:restriction base="mediumType">
            <xs:enumeration value="memory"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
```

```
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.20 SAML Schema认证关联 — SPKI

这是公开密钥 SAML 认证关联 Schema。SPKI 关联类别指示通过数字签字的方式对责任人进行的认证。数字签字使用的密钥是经过 SPKI 体系架构验证的。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI
        Document identifier: saml-schema-authn-context-spki-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
```

```
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:SPKI"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.21 SAML Schema认证关联 — SRP

下表列出 SRP[参见 IETF RFC 2945] SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier:
urn:oasis:names:tc:SAML:2.0:ac:classes:SecureRemotePassword
        Document identifier: saml-schema-authn-context-srp-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
```

```
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SharedSecretChallengeResponse"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="SharedSecretChallengeResponseType">
      <xs:complexContent>
        <xs:restriction base="SharedSecretChallengeResponseType">
          <xs:attribute name="method" type="xs:anyURI" fixed="urn:ietf:rfc:2945"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.22 SAML Schema认证关联 — Telephony

这是电话 SAML 认证关联 Schema，用于通过预协议子集的固定电话号码对责任人并通过电话协议进行传送进行认证。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:Telephony
        Document identifier: saml-schema-authn-context-telephony-2.0
```

```
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="SubscriberLineNumber"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
            <xs:choice>
              <xs:element ref="PSTN"/>
              <xs:element ref="ISDN"/>
              <xs:element ref="ADSL"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.23 SAML Schema认证关联 — TimeSync

这是 TimeSynchToken SAML 认证关联 Schema。通过时间同步令牌对责任人进行以上，TimeSyncToken 可应用。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:TimeSyncToken
        Document identifier: saml-schema-authn-context-timesync-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="Token"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="TokenType">
      <xs:complexContent>
        <xs:restriction base="TokenType">
          <xs:sequence>
            <xs:element ref="TimeSyncToken"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
```

```
    </xs:complexType>

    <xs:complexType name="TimeSyncTokenType">
      <xs:complexContent>
        <xs:restriction base="TimeSyncTokenType">
          <xs:attribute name="DeviceType" use="required">
            <xs:simpleType>
              <xs:restriction base="DeviceTypeType">
                <xs:enumeration value="hardware"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>

          <xs:attribute name="SeedLength" use="required">
            <xs:simpleType>
              <xs:restriction base="xs:integer">
                <xs:minInclusive value="64"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>

          <xs:attribute name="DeviceInHand" use="required">
            <xs:simpleType>
              <xs:restriction base="booleanType">
                <xs:enumeration value="true"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.24 SAML Schema认证关联类型

下表列出 SAML 认证关联类型 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  version="2.0">

  <xs:annotation>
    <xs:documentation>
      Document identifier: saml-schema-authn-context-types-2.0
      Location: http://docs.oasis-open.org/security/saml/v2.0/
      Revision history:
          V2.0 (March, 2005):
          New core authentication context schema types for SAML V2.0.
    </xs:documentation>
  </xs:annotation>

  <xs:element name="AuthenticationContextDeclaration"
type="AuthnContextDeclarationBaseType">
    <xs:annotation>
      <xs:documentation>
        A particular assertion u111 on an identity
        provider's part with respect to the authentication
        context associated with an authentication assertion.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="Identification" type="IdentificationType">
    <xs:annotation>
```

```
    <xs:documentation>
      Refers to those characteristics that describe the
      processes and mechanisms
      the Authentication Authority uses to initially create
      an association between_u97 ? Principal
      and the identity (or name) by which the Principal will
      be known
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="PhysicalVerification">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that identification has been
      performed in a physical
      face-to-face meeting with the principal and not in an
      online manner.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:attribute name="credentialLevel">
      <xs:simpleType>
        <xs:restriction base="xs:NMTOKEN">
          <xs:enumeration value="primary"/>
          <xs:enumeration value="secondary"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>

<xs:element name="WrittenConsent" type="ExtensionOnlyType"/>

<xs:element name="TechnicalProtection" type="TechnicalProtectionBaseType">
  <xs:annotation>
    <xs:documentation>
      Refers to those characterstics that describe how the
      'secret' (the knowledge or possession
      of which allows the Principal to authenticate to the
      Authentication Authority) is kept secure
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SecretKeyProtection" type="SecretKeyProtectionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 the types and strengths of
      facilities
      of a UA used to protect a shared secret key from
      unauthorized access and/or use.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="PrivateKeyProtection" type="PrivateKeyProtectionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 the types and strengths of
      facilities
      of a UA used to protect a private key from
      unauthorized access and/or use.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="KeyActivation" type="KeyActivationType">
  <xs:annotation>
    <xs:documentation>The actions that must be performed
      before the private key_u99 can be used. </xs:documentation>
```

```
  </xs:annotation>
</xs:element>

<xs:element name="KeySharing" type="KeySharingType">
  <xs:annotation>
    <xs:documentation>Whether or not the private key_u105 is shared
      with the certificate authority.</xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="KeyStorage" type="KeyStorageType">
  <xs:annotation>
    <xs:documentation>
      In which medium is the_u107 key stored.
      memory - the key is stored in memory.
      smartcard - the key is_u115 stored in a smartcard.
      token - the key is stored in a hardware token.
      MobileDevice - the key_u105 is stored in a mobile device.
      MobileAuthCard - the key is stored in a mobile
      authentication card.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SubscriberLineNumber" type="ExtensionOnlyType"/>
<xs:element name="UserSuffix" type="ExtensionOnlyType"/>

<xs:element name="Password" type="PasswordType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that a password (or passphrase)
      has been used to
      authenticate the Principal to a remote system.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationPin" type="ActivationPinType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that a Pin (Personal
      Identification Number)_u104 has been used to authenticate the Principal
      to some local system in order to activate a key.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Token" type="TokenType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that a hardware or software
      token is used
      as a method of identifying the Principal.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="TimeSyncToken" type="TimeSyncTokenType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that a time synchronization
      token is used to identify the Principal. hardware -
      the time synchonization
      token has been implemented in hardware. software - the
      time synchronization
      token has been implemented in software. SeedLength -
      the length, in bits, of the
      random seed used in the time synchronization token.
    </xs:documentation>
  </xs:annotation>
</xs:element>
```

```
<xs:element name="Smartcard" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that a smartcard is used to
      identify the Principal.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Length" type="LengthType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 the minimum and/or maximum
      ASCII length of the password which is enforced (by the UA or the
      IdP). In other words, this is the minimum and/or maximum number of
      ASCII characters required to represent a valid password.
      min - the minimum number of ASCII characters required
      in a valid password, as enforced by the UA or the IdP.
      max - the maximum number of ASCII characters required
      in a valid password, as enforced by the UA or the IdP.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationLimit" type="ActivationLimitType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 the length of time for which an
      PIN-based authentication is valid.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Generation">
  <xs:annotation>
    <xs:documentation>
      Indicates whether the password was chosen by the
      Principal or auto-supplied by the Authentication Authority.
      principal chosen - the Principal is allowed to choose
      the value of the password. This is true even if
      the initial password is chosen at random by the UA or
      the IdP and the Principal is then free to change
      the password.
      automatic - the password is chosen by the UA or the
      IdP to be cryptographically strong in some sense,
      or to satisfy certain password rules, and that the
      Principal is not free to change it or to choose a new password.
    </xs:documentation>
  </xs:annotation>

  <xs:complexType>
    <xs:attribute name="mechanism" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:NMTOKEN">
          <xs:enumeration value="principalchosen"/>
          <xs:enumeration value="automatic"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>

<xs:element name="AuthnMethod" type="AuthnMethodBaseType">
  <xs:annotation>
    <xs:documentation>
      Refers to those characteristics that define the
      mechanisms by which the Principal authenticates to the Authentication
      Authority.
    </xs:documentation>
  </xs:annotation>
```

```
  </xs:element>

  <xs:element name="PrincipalAuthenticationMechanism"
type="PrincipalAuthenticationMechanismType">
    <xs:annotation>
      <xs:documentation>
        The method that a Principal employs to perform
        authentication to local system components.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="Authenticator" type="AuthenticatorBaseType">
    <xs:annotation>
      <xs:documentation>
        The method applied to validate a principal's
        authentication across a network
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ComplexAuthenticator" type="ComplexAuthenticatorType">
    <xs:annotation>
      <xs:documentation>
        Supports Authenticators with nested combinations of
        additional complexity.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="PreviousSession" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        Indicates that the Principal has been strongly
        authenticated in a previous session during which the IdP has set a
        cookie in the UA. During the present session the Principal has only
        been authenticated by the UA returning the cookie to the IdP.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ResumeSession" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        Rather like PreviousSession but using stronger
        security. A secret that was established in a previous session with
        the Authentication Authority has been cached by the local system and
        is now re-used (e.g. a_u77 master Secret is used to derive new session
        keys in TLS, SSL, WTLS).
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ZeroKnowledge" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates_u116 that the Principal has been
        authenticated by a zero knowledge technique as specified in ISO/IEC
        9798-5.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="SharedSecretChallengeResponse"
type="SharedSecretChallengeResponseType"/>

  <xs:complexType name="SharedSecretChallengeResponseType">
    <xs:annotation>
      <xs:documentation>
        This element indicates_u116 that the Principal has been
        authenticated by a challenge-response protocol utilizing shared secret
```

```
        keys and symmetric cryptography.
      </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="method" type="xs:anyURI" use="optional"/>
  </xs:complexType>

  <xs:element name="DigSig" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates_u116 that the Principal has been
        authenticated by a mechanism which involves the Principal computing a
        digital signature over_u97 at least challenge data provided by the IdP.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AsymmetricDecryption" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        The local system has a_u112 Private key but it is used
        in decryption mode, rather than signature mode. For example, the
        Authentication Authority generates a secret and encrypts it using the
        local system's public key: the local system then proves it has
        decrypted the secret.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AsymmetricKeyAgreement" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        The local system has a_u112 Private key and uses it for
        shared secret key agreement with the Authentication Authority (e.g.
        via Diffie Helman).
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:complexType name="PublicKeyType">
    <xs:sequence>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="keyValidation" use="optional"/>
  </xs:complexType>

  <xs:element name="IPAddress" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates_u116 that the Principal has been
        authenticated through connection from a particular IP address.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="SharedSecretDynamicPlaintext" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        The local system and Authentication Authority
        share a secret key. The local system uses this to encrypt a
        randomised string to pass to the Authentication Authority.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AuthenticatorTransportProtocol"
type="AuthenticatorTransportProtocolType">
    <xs:annotation>
      <xs:documentation>
```

```
      The protocol across which Authenticator information is
      transferred to an Authentication Authority verifier.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="HTTP" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Authenticator has been
      transmitted using bare_u72 HTTP utilizing no additional security
      protocols.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="IPSec" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Authenticator has been
      transmitted using a transport mechanism protected by an IPSEC session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="WTLS" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Authenticator has been
      transmitted using a transport mechanism protected by a WTLS session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="MobileNetworkNoEncryption" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Authenticator has been
      transmitted solely across a mobile network using no additional
      security mechanism.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="MobileNetworkRadioEncryption" type="ExtensionOnlyType"/>
<xs:element name="MobileNetworkEndToEndEncryption" type="ExtensionOnlyType"/>

<xs:element name="SSL" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Authenticator has been
      transmitted using a transport mechnanism protected by an SSL or TLS
      session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="PSTN" type="ExtensionOnlyType"/>
<xs:element name="ISDN" type="ExtensionOnlyType"/>
<xs:element name="ADSL" type="ExtensionOnlyType"/>

<xs:element name="OperationalProtection" type="OperationalProtectionType">
  <xs:annotation>
    <xs:documentation>
      Refers to those characteristics that describe
      procedural security controls employed by the Authentication Authority.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SecurityAudit" type="SecurityAuditType"/>
```

```
<xs:element name="SwitchAudit" type="ExtensionOnlyType"/>
<xs:element name="DeactivationCallCenter" type="ExtensionOnlyType"/>

<xs:element name="GoverningAgreements" type="GoverningAgreementsType">
  <xs:annotation>
    <xs:documentation>
      Provides a mechanism for linking to external (likely
      human readable) documents in which additional business agreements,
      (e.g. liability constraints, obligations, etc.) can be placed.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="GoverningAgreementRef" type="GoverningAgreementRefType"/>

<xs:simpleType name="nymType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="anonymity"/>
    <xs:enumeration value="verinymity"/>
    <xs:enumeration value="pseudonymity"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="AuthnContextDeclarationBaseType">
  <xs:sequence>
    <xs:element ref="Identification" minOccurs="0"/>
    <xs:element ref="TechnicalProtection" minOccurs="0"/>
    <xs:element ref="OperationalProtection" minOccurs="0"/>
    <xs:element ref="AuthnMethod" minOccurs="0"/>
    <xs:element ref="GoverningAgreements" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="ID" type="xs:ID" use="optional"/>
</xs:complexType>

<xs:complexType name="IdentificationType">
  <xs:sequence>
    <xs:element ref="PhysicalVerification" minOccurs="0"/>
    <xs:element ref="WrittenConsent" minOccurs="0"/>
    <xs:element ref="GoverningAgreements" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="nym" type="nymType">
    <xs:annotation>
      <xs:documentation>
        This attribute indicates whether or not the
        Identification mechanisms allow the actions of the Principal to be
        linked to an actual end user.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>

<xs:complexType name="TechnicalProtectionBaseType">
  <xs:sequence>
    <xs:choice minOccurs="0">
      <xs:element ref="PrivateKeyProtection"/>
      <xs:element ref="SecretKeyProtection"/>
    </xs:choice>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="OperationalProtectionType">
  <xs:sequence>
    <xs:element ref="SecurityAudit" minOccurs="0"/>
    <xs:element ref="DeactivationCallCenter" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="AuthnMethodBaseType">
  <xs:sequence>
    <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
    <xs:element ref="Authenticator" minOccurs="0"/>
    <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="GoverningAgreementsType">
  <xs:sequence>
    <xs:element ref="GoverningAgreementRef" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="GoverningAgreementRefType">
  <xs:attribute name="governingAgreementRef" type="xs:anyURI" use="required"/>
</xs:complexType>

<xs:complexType name="PrincipalAuthenticationMechanismType">
  <xs:sequence>
    <xs:element ref="Password" minOccurs="0"/>
    <xs:element ref="RestrictedPassword" minOccurs="0"/>
    <xs:element ref="Token" minOccurs="0"/>
    <xs:element ref="Smartcard" minOccurs="0"/>
    <xs:element ref="ActivationPin" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="preauth" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:group name="AuthenticatorChoiceGroup">
  <xs:choice>
    <xs:element ref="PreviousSession"/>
    <xs:element ref="ResumeSession"/>
    <xs:element ref="DigSig"/>
    <xs:element ref="Password"/>
    <xs:element ref="RestrictedPassword"/>
    <xs:element ref="ZeroKnowledge"/>
    <xs:element ref="SharedSecretChallengeResponse"/>
    <xs:element ref="SharedSecretDynamicPlaintext"/>
    <xs:element ref="IPAddress"/>
    <xs:element ref="AsymmetricDecryption"/>
    <xs:element ref="AsymmetricKeyAgreement"/>
    <xs:element ref="SubscriberLineNumber"/>
    <xs:element ref="UserSuffix"/>
    <xs:element ref="ComplexAuthenticator"/>
  </xs:choice>
</xs:group>

<xs:group name="AuthenticatorSequenceGroup">
  <xs:sequence>
    <xs:element ref="PreviousSession" minOccurs="0"/>
    <xs:element ref="ResumeSession" minOccurs="0"/>
    <xs:element ref="DigSig" minOccurs="0"/>
    <xs:element ref="Password" minOccurs="0"/>
    <xs:element ref="RestrictedPassword" minOccurs="0"/>
    <xs:element ref="ZeroKnowledge" minOccurs="0"/>
    <xs:element ref="SharedSecretChallengeResponse" minOccurs="0"/>
    <xs:element ref="SharedSecretDynamicPlaintext" minOccurs="0"/>
    <xs:element ref="IPAddress" minOccurs="0"/>
    <xs:element ref="AsymmetricDecryption" minOccurs="0"/>
    <xs:element ref="AsymmetricKeyAgreement" minOccurs="0"/>
    <xs:element ref="SubscriberLineNumber" minOccurs="0"/>
    <xs:element ref="UserSuffix" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="AuthenticatorBaseType">
  <xs:sequence>
```

```
    <xs:group ref="AuthenticatorChoiceGroup"/>
    <xs:group ref="AuthenticatorSequenceGroup"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ComplexAuthenticatorType">
  <xs:sequence>
    <xs:group ref="AuthenticatorChoiceGroup"/>
    <xs:group ref="AuthenticatorSequenceGroup"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="AuthenticatorTransportProtocolType">
  <xs:sequence>
    <xs:choice minOccurs="0">
      <xs:element ref="HTTP"/>
      <xs:element ref="SSL"/>
      <xs:element ref="MobileNetworkNoEncryption"/>
      <xs:element ref="MobileNetworkRadioEncryption"/>
      <xs:element ref="MobileNetworkEndToEndEncryption"/>
      <xs:element ref="WTLS"/>
      <xs:element ref="IPSec"/>
      <xs:element ref="PSTN"/>
      <xs:element ref="ISDN"/>
      <xs:element ref="ADSL"/>
    </xs:choice>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="KeyActivationType">
  <xs:sequence>
    <xs:element ref="ActivationPin" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="KeySharingType">
  <xs:attribute name="sharing" type="xs:boolean" use="required"/>
</xs:complexType>

<xs:complexType name="PrivateKeyProtectionType">
  <xs:sequence>
    <xs:element ref="KeyActivation" minOccurs="0"/>
    <xs:element ref="KeyStorage" minOccurs="0"/>
    <xs:element ref="KeySharing" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PasswordType">
  <xs:sequence>
    <xs:element ref="Length" minOccurs="0"/>
    <xs:element ref="Alphabet" minOccurs="0"/>
    <xs:element ref="Generation" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="ExternalVerification" type="xs:anyURI" use="optional"/>
</xs:complexType>

<xs:element name="RestrictedPassword" type="RestrictedPasswordType"/>

<xs:complexType name="RestrictedPasswordType">
  <xs:complexContent>
    <xs:restriction base="PasswordType">
      <xs:sequence>
        <xs:element name="Length" type="RestrictedLengthType" minOccurs="1"/>
        <xs:element ref="Generation" minOccurs="0"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
```

```
        <xs:attribute name="ExternalVerification" type="xs:anyURI"
use="optional"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="RestrictedLengthType">
    <xs:complexContent>
      <xs:restriction base="LengthType">
        <xs:attribute name="min" use="required">
          <xs:simpleType>
            <xs:restriction base="xs:integer">
              <xs:minInclusive value="3"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="max" type="xs:integer" use="optional"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="ActivationPinType">
    <xs:sequence>
      <xs:element ref="Length" minOccurs="0"/>
      <xs:element ref="Alphabet" minOccurs="0"/>
      <xs:element ref="Generation" minOccurs="0"/>
      <xs:element ref="ActivationLimit" minOccurs="0"/>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="Alphabet" type="AlphabetType"/>
  <xs:complexType name="AlphabetType">
    <xs:attribute name="requiredChars" type="xs:string" use="required"/>
    <xs:attribute name="excludedChars" type="xs:string" use="optional"/>
    <xs:attribute name="case" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:complexType name="TokenType">
    <xs:sequence>
      <xs:element ref="TimeSyncToken"/>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="DeviceTypeType">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="hardware"/>
      <xs:enumeration value="software"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="booleanType">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="true"/>
      <xs:enumeration value="false"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="TimeSyncTokenType">
    <xs:attribute name="DeviceType" type="DeviceTypeType" use="required"/>
    <xs:attribute name="SeedLength" type="xs:integer" use="required"/>
    <xs:attribute name="DeviceInHand" type="booleanType" use="required"/>
  </xs:complexType>

  <xs:complexType name="ActivationLimitType">
    <xs:choice>
      <xs:element ref="ActivationLimitDuration"/>
      <xs:element ref="ActivationLimitUsages"/>
      <xs:element ref="ActivationLimitSession"/>
    </xs:choice>
```

```
</xs:complexType>

<xs:element name="ActivationLimitDuration" type="ActivationLimitDurationType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Key Activation Limit is
      defined as a specific duration of time.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationLimitUsages" type="ActivationLimitUsagesType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Key Activation Limit is
      defined as a number of_u117 usages.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationLimitSession" type="ActivationLimitSessionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates_u116 that the Key Activation Limit is
      the session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:complexType name="ActivationLimitDurationType">
  <xs:attribute name="duration" type="xs:duration" use="required"/>
</xs:complexType>

<xs:complexType name="ActivationLimitUsagesType">
  <xs:attribute name="number" type="xs:integer" use="required"/>
</xs:complexType>

<xs:complexType name="ActivationLimitSessionType"/>

<xs:complexType name="LengthType">
  <xs:attribute name="min" type="xs:integer" use="required"/>
  <xs:attribute name="max" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:simpleType name="mediumType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="memory"/>
    <xs:enumeration value="smartcard"/>
    <xs:enumeration value="token"/>
    <xs:enumeration value="MobileDevice"/>
    <xs:enumeration value="MobileAuthCard"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="KeyStorageType">
  <xs:attribute name="medium" type="mediumType" use="required"/>
</xs:complexType>

<xs:complexType name="SecretKeyProtectionType">
  <xs:sequence>
    <xs:element ref="KeyActivation" minOccurs="0"/>
    <xs:element ref="KeyStorage" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="SecurityAuditType">
  <xs:sequence>
    <xs:element ref="SwitchAudit" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
```

```
  </xs:complexType>

  <xs:complexType name="ExtensionOnlyType">
    <xs:sequence>
      <xs:element ref="Extension" minOccurs="0"  maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="Extension" type="ExtensionType"/>

  <xs:complexType name="ExtensionType">
    <xs:sequence>
      <xs:any namespace="##other" processContents="lax" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

</xs:schema>
```

A.25 SAML Schema认证关联 — X.509

下表列出 X.509 SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:X509
        Document identifier: saml-schema-authn-context-x509-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
```

```
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.26 SAML Schema认证关联 — XMLDSig

下表列出 XML 数字签字 SAML 认证关联 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:XMLDSig
        Document identifier: saml-schema-authn-context-xmldsig-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
```

```
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:ietf:rfc:3075"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

A.27 SAML Schema — ECP

下面列出增强的客户端或者代理（ECP）用户数据 SAML Schema 列表。

```
<?xml version="1.0" encoding="UTF-8"?>
<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>
```

A.28 SAML Schema元数据

下表列出 SAML 元数据 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<schema
    targetNamespace="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="http://www.w3.org/2000/09/xmldsig#"

schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/xmldsig-core
-schema.xsd"/>
    <import namespace="http://www.w3.org/2001/04/xmlenc#"

schemaLocation="http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/xenc-schema.x
sd"/>
    <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
        schemaLocation="saml-schema-assertion-2.0.xsd"/>
    <import namespace="http://www.w3.org/XML/1998/namespace"
        schemaLocation="http://www.w3.org/2001/xml.xsd"/>
    <annotation>
        <documentation>
            Document identifier: saml-schema-metadata-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
              V2.0 (March, 2005):
                Schema for SAML metadata, first published in SAML 2.0.
        </documentation>
    </annotation>

    <simpleType name="entityIDType">
        <restriction base="anyURI">
            <maxLength value="1024"/>
        </restriction>
    </simpleType>
    <complexType name="localizedNameType">
        <simpleContent>
            <extension base="string">
                <attribute ref="xml:lang" use="required"/>
            </extension>
        </simpleContent>
    </complexType>
    <complexType name="localizedURIType">
        <simpleContent>
            <extension base="anyURI">
                <attribute ref="xml:lang" use="required"/>
            </extension>
        </simpleContent>
    </complexType>

    <element name="Extensions" type="md:ExtensionsType"/>
    <complexType final="#all" name="ExtensionsType">
        <sequence>
            <any namespace="##other" processContents="lax" maxOccurs="unbounded"/>
        </sequence>
    </complexType>

    <complexType name="EndpointType">
        <sequence>
            <any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
```

```
        <attribute name="Binding" type="anyURI" use="required"/>
        <attribute name="Location" type="anyURI" use="required"/>
        <attribute name="ResponseLocation" type="anyURI" use="optional"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>

    <complexType name="IndexedEndpointType">
        <complexContent>
            <extension base="md:EndpointType">
                <attribute name="index" type="unsignedShort" use="required"/>
                <attribute name="isDefault" type="boolean" use="optional"/>
            </extension>
        </complexContent>
    </complexType>

    <element name="EntitiesDescriptor" type="md:EntitiesDescriptorType"/>
    <complexType name="EntitiesDescriptorType">
        <sequence>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="md:Extensions" minOccurs="0"/>
            <choice minOccurs="1" maxOccurs="unbounded">
                <element ref="md:EntityDescriptor"/>
                <element ref="md:EntitiesDescriptor"/>
            </choice>
        </sequence>
        <attribute name="validUntil" type="dateTime" use="optional"/>
        <attribute name="cacheDuration" type="duration" use="optional"/>
        <attribute name="ID" type="ID" use="optional"/>
        <attribute name="Name" type="string" use="optional"/>
    </complexType>

    <element name="EntityDescriptor" type="md:EntityDescriptorType"/>
    <complexType name="EntityDescriptorType">
        <sequence>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="md:Extensions" minOccurs="0"/>
            <choice>
                <choice maxOccurs="unbounded">
                    <element ref="md:RoleDescriptor"/>
                    <element ref="md:IDPSSODescriptor"/>
                    <element ref="md:SPSSODescriptor"/>
                    <element ref="md:AuthnAuthorityDescriptor"/>
                    <element ref="md:AttributeAuthorityDescriptor"/>
                    <element ref="md:PDPDescriptor"/>
                </choice>
                <element ref="md:AffiliationDescriptor"/>
            </choice>
            <element ref="md:Organization" minOccurs="0"/>
            <element ref="md:ContactPerson" minOccurs="0" maxOccurs="unbounded"/>
            <element ref="md:AdditionalMetadataLocation" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
        <attribute name="entityID" type="md:entityIDType" use="required"/>
        <attribute name="validUntil" type="dateTime" use="optional"/>
        <attribute name="cacheDuration" type="duration" use="optional"/>
        <attribute name="ID" type="ID" use="optional"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>

    <element name="Organization" type="md:OrganizationType"/>
    <complexType name="OrganizationType">
        <sequence>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:OrganizationName" maxOccurs="unbounded"/>
            <element ref="md:OrganizationDisplayName" maxOccurs="unbounded"/>
            <element ref="md:OrganizationURL" maxOccurs="unbounded"/>
        </sequence>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>
    <element name="OrganizationName" type="md:localizedNameType"/>
    <element name="OrganizationDisplayName" type="md:localizedNameType"/>
```

```
    <element name="OrganizationURL" type="md:localizedURIType"/>
    <element name="ContactPerson" type="md:ContactType"/>
    <complexType name="ContactType">
        <sequence>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:Company" minOccurs="0"/>
            <element ref="md:GivenName" minOccurs="0"/>
            <element ref="md:SurName" minOccurs="0"/>
            <element ref="md:EmailAddress" minOccurs="0" maxOccurs="unbounded"/>
            <element ref="md:TelephoneNumber" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
        <attribute name="contactType" type="md:ContactTypeType" use="required"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>
    <element name="Company" type="string"/>
    <element name="GivenName" type="string"/>
    <element name="SurName" type="string"/>
    <element name="EmailAddress" type="anyURI"/>
    <element name="TelephoneNumber" type="string"/>
    <simpleType name="ContactTypeType">
        <restriction base="string">
            <enumeration value="technical"/>
            <enumeration value="support"/>
            <enumeration value="administrative"/>
            <enumeration value="billing"/>
            <enumeration value="other"/>
        </restriction>
    </simpleType>

    <element name="AdditionalMetadataLocation"
type="md:AdditionalMetadataLocationType"/>
    <complexType name="AdditionalMetadataLocationType">
        <simpleContent>
            <extension base="anyURI">
                <attribute name="namespace" type="anyURI" use="required"/>
            </extension>
        </simpleContent>
    </complexType>

    <element name="RoleDescriptor" type="md:RoleDescriptorType"/>
    <complexType name="RoleDescriptorType" abstract="true">
        <sequence>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:KeyDescriptor" minOccurs="0" maxOccurs="unbounded"/>
            <element ref="md:Organization" minOccurs="0"/>
            <element ref="md:ContactPerson" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
        <attribute name="ID" type="ID" use="optional"/>
        <attribute name="validUntil" type="dateTime" use="optional"/>
        <attribute name="cacheDuration" type="duration" use="optional"/>
        <attribute name="protocolSupportEnumeration" type="md:anyURIListType"
use="required"/>
        <attribute name="errorURL" type="anyURI" use="optional"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>
    <simpleType name="anyURIListType">
        <list itemType="anyURI"/>
    </simpleType>

    <element name="KeyDescriptor" type="md:KeyDescriptorType"/>
    <complexType name="KeyDescriptorType">
        <sequence>
            <element ref="ds:KeyInfo"/>
            <element ref="md:EncryptionMethod" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
        <attribute name="use" type="md:KeyTypes" use="optional"/>
    </complexType>
    <simpleType name="KeyTypes">
```

```
        <restriction base="string">
            <enumeration value="encryption"/>
            <enumeration value="signing"/>
        </restriction>
    </simpleType>
    <element name="EncryptionMethod" type="xenc:EncryptionMethodType"/>

    <complexType name="SSODescriptorType" abstract="true">
        <complexContent>
            <extension base="md:RoleDescriptorType">
                <sequence>
                    <element ref="md:ArtifactResolutionService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:SingleLogoutService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:ManageNameIDService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="ArtifactResolutionService" type="md:IndexedEndpointType"/>
    <element name="SingleLogoutService" type="md:EndpointType"/>
    <element name="ManageNameIDService" type="md:EndpointType"/>
    <element name="NameIDFormat" type="anyURI"/>

    <element name="IDPSSODescriptor" type="md:IDPSSODescriptorType"/>
    <complexType name="IDPSSODescriptorType">
        <complexContent>
            <extension base="md:SSODescriptorType">
                <sequence>
                    <element ref="md:SingleSignOnService" maxOccurs="unbounded"/>
                    <element ref="md:NameIDMappingService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:AssertionIDRequestService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:AttributeProfile" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
                <attribute name="WantAuthnRequestsSigned" type="boolean"
use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="SingleSignOnService" type="md:EndpointType"/>
    <element name="NameIDMappingService" type="md:EndpointType"/>
    <element name="AssertionIDRequestService" type="md:EndpointType"/>
    <element name="AttributeProfile" type="anyURI"/>

    <element name="SPSSODescriptor" type="md:SPSSODescriptorType"/>
    <complexType name="SPSSODescriptorType">
        <complexContent>
            <extension base="md:SSODescriptorType">
                <sequence>
                    <element ref="md:AssertionConsumerService"
maxOccurs="unbounded"/>
                    <element ref="md:AttributeConsumingService" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
                <attribute name="AuthnRequestsSigned" type="boolean"
use="optional"/>
                <attribute name="WantAssertionsSigned" type="boolean"
use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="AssertionConsumerService" type="md:IndexedEndpointType"/>
```

```
    <element name="AttributeConsumingService"
type="md:AttributeConsumingServiceType"/>
    <complexType name="AttributeConsumingServiceType">
        <sequence>
            <element ref="md:ServiceName" maxOccurs="unbounded"/>
            <element ref="md:ServiceDescription" minOccurs="0"
maxOccurs="unbounded"/>
            <element ref="md:RequestedAttribute" maxOccurs="unbounded"/>
        </sequence>
        <attribute name="index" type="unsignedShort" use="required"/>
        <attribute name="isDefault" type="boolean" use="optional"/>
    </complexType>
    <element name="ServiceName" type="md:localizedNameType"/>
    <element name="ServiceDescription" type="md:localizedNameType"/>
    <element name="RequestedAttribute" type="md:RequestedAttributeType"/>
    <complexType name="RequestedAttributeType">
        <complexContent>
            <extension base="saml:AttributeType">
                <attribute name="isRequired" type="boolean" use="optional"/>
            </extension>
        </complexContent>
    </complexType>

    <element name="AuthnAuthorityDescriptor"
type="md:AuthnAuthorityDescriptorType"/>
    <complexType name="AuthnAuthorityDescriptorType">
        <complexContent>
            <extension base="md:RoleDescriptorType">
                <sequence>
                    <element ref="md:AuthnQueryService" maxOccurs="unbounded"/>
                    <element ref="md:AssertionIDRequestService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="AuthnQueryService" type="md:EndpointType"/>

    <element name="PDPDescriptor" type="md:PDPDescriptorType"/>
    <complexType name="PDPDescriptorType">
        <complexContent>
            <extension base="md:RoleDescriptorType">
                <sequence>
                    <element ref="md:AuthzService" maxOccurs="unbounded"/>
                    <element ref="md:AssertionIDRequestService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="AuthzService" type="md:EndpointType"/>

    <element name="AttributeAuthorityDescriptor"
type="md:AttributeAuthorityDescriptorType"/>
    <complexType name="AttributeAuthorityDescriptorType">
        <complexContent>
            <extension base="md:RoleDescriptorType">
                <sequence>
                    <element ref="md:AttributeService" maxOccurs="unbounded"/>
                    <element ref="md:AssertionIDRequestService" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:NameIDFormat" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="md:AttributeProfile" minOccurs="0"
maxOccurs="unbounded"/>
                    <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
```

```
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="AttributeService" type="md:EndpointType"/>

    <element name="AffiliationDescriptor" type="md:AffiliationDescriptorType"/>
    <complexType name="AffiliationDescriptorType">
        <sequence>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="md:Extensions" minOccurs="0"/>
            <element ref="md:AffiliateMember" maxOccurs="unbounded"/>
            <element ref="md:KeyDescriptor" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
        <attribute name="affiliationOwnerID" type="md:entityIDType"
use="required"/>
        <attribute name="validUntil" type="dateTime" use="optional"/>
        <attribute name="cacheDuration" type="duration" use="optional"/>
        <attribute name="ID" type="ID" use="optional"/>
        <anyAttribute namespace="##other" processContents="lax"/>
    </complexType>
    <element name="AffiliateMember" type="md:entityIDType"/>
</schema>
```

A.29 SAML Schema协议

下表列出 SAML 协议 Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<schema
    targetNamespace="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <import namespace="urn:oasis:names:tc:SAML:2.0:assertion"
        schemaLocation="saml-schema-assertion-2.0.xsd"/>
    <import namespace="http://www.w3.org/2000/09/xmldsig#"

schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/xmldsig-core
-schema.xsd"/>
    <annotation>
        <documentation>
            Document identifier: saml-schema-protocol-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V1.0 (November, 2002):
              Initial Standard_u83 schema.
            V1.1 (September, 2003):
              Updates within the same V1.0 namespace.
            V2.0 (March, 2005):
              New protocol schema based in a SAML V2.0 namespace.
     </documentation>
    </annotation>
    <complexType name="RequestAbstractType" abstract="true">
        <sequence>
            <element ref="saml:Issuer" minOccurs="0"/>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="samlp:Extensions" minOccurs="0"/>
        </sequence>
        <attribute name="ID" type="ID" use="required"/>
        <attribute name="Version" type="string" use="required"/>
        <attribute name="IssueInstant" type="dateTime" use="required"/>
        <attribute name="Destination" type="anyURI" use="optional"/>
         <attribute name="Consent" type="anyURI" use="optional"/>
```

```
    </complexType>
    <element name="Extensions" type="samlp:ExtensionsType"/>
    <complexType name="ExtensionsType">
        <sequence>
            <any namespace="##other" processContents="lax" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <complexType name="StatusResponseType">
         <sequence>
            <element ref="saml:Issuer" minOccurs="0"/>
            <element ref="ds:Signature" minOccurs="0"/>
            <element ref="samlp:Extensions" minOccurs="0"/>
            <element ref="samlp:Status"/>
         </sequence>
         <attribute name="ID" type="ID" use="required"/>
         <attribute name="InResponseTo" type="NCName" use="optional"/>
         <attribute name="Version" type="string" use="required"/>
         <attribute name="IssueInstant" type="dateTime" use="required"/>
         <attribute name="Destination" type="anyURI" use="optional"/>
         <attribute name="Consent" type="anyURI" use="optional"/>
    </complexType>
    <element name="Status" type="samlp:StatusType"/>
    <complexType name="StatusType">
        <sequence>
            <element ref="samlp:StatusCode"/>
            <element ref="samlp:StatusMessage" minOccurs="0"/>
            <element ref="samlp:StatusDetail" minOccurs="0"/>
        </sequence>
    </complexType>
    <element name="StatusCode" type="samlp:StatusCodeType"/>
    <complexType name="StatusCodeType">
        <sequence>
            <element ref="samlp:StatusCode" minOccurs="0"/>
        </sequence>
        <attribute name="Value" type="anyURI" use="required"/>
    </complexType>
    <element name="StatusMessage" type="string"/>
    <element name="StatusDetail" type="samlp:StatusDetailType"/>
    <complexType name="StatusDetailType">
        <sequence>
            <any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <element name="AssertionIDRequest" type="samlp:AssertionIDRequestType"/>
    <complexType name="AssertionIDRequestType">
         <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <element ref="saml:AssertionIDRef" maxOccurs="unbounded"/>
                </sequence>
            </extension>
         </complexContent>
    </complexType>
    <element name="SubjectQuery" type="samlp:SubjectQueryAbstractType"/>
    <complexType name="SubjectQueryAbstractType" abstract="true">
         <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <element ref="saml:Subject"/>
                </sequence>
            </extension>
         </complexContent>
    </complexType>
    <element name="AuthnQuery" type="samlp:AuthnQueryType"/>
    <complexType name="AuthnQueryType">
        <complexContent>
            <extension base="samlp:SubjectQueryAbstractType">
                <sequence>
                    <element ref="samlp:RequestedAuthnContext" minOccurs="0"/>
                </sequence>
```

```
                <attribute name="SessionIndex" type="string" use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="RequestedAuthnContext"
type="samlp:RequestedAuthnContextType"/>
    <complexType name="RequestedAuthnContextType">
        <choice>
            <element ref="saml:AuthnContextClassRef" maxOccurs="unbounded"/>
            <element ref="saml:AuthnContextDeclRef" maxOccurs="unbounded"/>
        </choice>
        <attribute name="Comparison" type="samlp:AuthnContextComparisonType"
use="optional"/>
    </complexType>
    <simpleType name="AuthnContextComparisonType">
        <restriction base="string">
            <enumeration value="exact"/>
            <enumeration value="minimum"/>
            <enumeration value="maximum"/>
            <enumeration value="better"/>
        </restriction>
    </simpleType>
    <element name="AttributeQuery" type="samlp:AttributeQueryType"/>
    <complexType name="AttributeQueryType">
        <complexContent>
            <extension base="samlp:SubjectQueryAbstractType">
                <sequence>
                    <element ref="saml:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="AuthzDecisionQuery" type="samlp:AuthzDecisionQueryType"/>
    <complexType name="AuthzDecisionQueryType">
        <complexContent>
            <extension base="samlp:SubjectQueryAbstractType">
                <sequence>
                    <element ref="saml:Action" maxOccurs="unbounded"/>
                    <element ref="saml:Evidence" minOccurs="0"/>
                </sequence>
                <attribute name="Resource" type="anyURI" use="required"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="AuthnRequest" type="samlp:AuthnRequestType"/>
    <complexType name="AuthnRequestType">
        <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <element ref="saml:Subject" minOccurs="0"/>
                    <element ref="samlp:NameIDPolicy" minOccurs="0"/>
                    <element ref="saml:Conditions" minOccurs="0"/>
                    <element ref="samlp:RequestedAuthnContext" minOccurs="0"/>
                    <element ref="samlp:Scoping" minOccurs="0"/>
                </sequence>
                <attribute name="ForceAuthn" type="boolean" use="optional"/>
                <attribute name="IsPassive" type="boolean" use="optional"/>
                <attribute name="ProtocolBinding" type="anyURI" use="optional"/>
                <attribute name="AssertionConsumerServiceIndex"
type="unsignedShort" use="optional"/>
                <attribute name="AssertionConsumerServiceURL" type="anyURI"
use="optional"/>
                <attribute name="AttributeConsumingServiceIndex"
type="unsignedShort" use="optional"/>
                <attribute name="ProviderName" type="string" use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="NameIDPolicy" type="samlp:NameIDPolicyType"/>
    <complexType name="NameIDPolicyType">
```

```
        <attribute name="Format" type="anyURI" use="optional"/>
        <attribute name="SPNameQualifier" type="string" use="optional"/>
        <attribute name="AllowCreate" type="boolean" use="optional"/>
    </complexType>
    <element name="Scoping" type="samlp:ScopingType"/>
    <complexType name="ScopingType">
        <sequence>
            <element ref="samlp:IDPList" minOccurs="0"/>
            <element ref="samlp:RequesterID" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
        <attribute name="ProxyCount" type="nonNegativeInteger" use="optional"/>
    </complexType>
    <element name="RequesterID" type="anyURI"/>
    <element name="IDPList" type="samlp:IDPListType"/>
    <complexType name="IDPListType">
        <sequence>
            <element ref="samlp:IDPEntry" maxOccurs="unbounded"/>
            <element ref="samlp:GetComplete" minOccurs="0"/>
        </sequence>
    </complexType>
    <element name="IDPEntry" type="samlp:IDPEntryType"/>
    <complexType name="IDPEntryType">
        <attribute name="ProviderID" type="anyURI" use="required"/>
        <attribute name="Name" type="string" use="optional"/>
        <attribute name="Loc" type="anyURI" use="optional"/>
    </complexType>
    <element name="GetComplete" type="anyURI"/>
    <element name="Response" type="samlp:ResponseType"/>
    <complexType name="ResponseType">
         <complexContent>
            <extension base="samlp:StatusResponseType">
                <choice minOccurs="0" maxOccurs="unbounded">
                    <element ref="saml:Assertion"/>
                    <element ref="saml:EncryptedAssertion"/>
                </choice>
            </extension>
         </complexContent>
    </complexType>
    <element name="ArtifactResolve" type="samlp:ArtifactResolveType"/>
    <complexType name="ArtifactResolveType">
         <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <element ref="samlp:Artifact"/>
                </sequence>
            </extension>
         </complexContent>
    </complexType>
    <element name="Artifact" type="string"/>
    <element name="ArtifactResponse" type="samlp:ArtifactResponseType"/>
    <complexType name="ArtifactResponseType">
         <complexContent>
            <extension base="samlp:StatusResponseType">
                <sequence>
                    <any namespace="##any" processContents="lax" minOccurs="0"/>
                </sequence>
            </extension>
         </complexContent>
    </complexType>
    <element name="ManageNameIDRequest" type="samlp:ManageNameIDRequestType"/>
    <complexType name="ManageNameIDRequestType">
         <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <choice>
                        <element ref="saml:NameID"/>
                        <element ref="saml:EncryptedID"/>
                    </choice>
                    <choice>
                        <element ref="samlp:NewID"/>
                        <element ref="samlp:NewEncryptedID"/>
```

```
                        <element ref="samlp:Terminate"/>
                    </choice>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NewID" type="string"/>
    <element name="NewEncryptedID" type="saml:EncryptedElementType"/>
    <element name="Terminate" type="samlp:TerminateType"/>
    <complexType name="TerminateType"/>
    <element name="ManageNameIDResponse" type="samlp:StatusResponseType"/>
    <element name="LogoutRequest" type="samlp:LogoutRequestType"/>
    <complexType name="LogoutRequestType">
        <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <choice>
                        <element ref="saml:BaseID"/>
                        <element ref="saml:NameID"/>
                        <element ref="saml:EncryptedID"/>
                    </choice>
                    <element ref="samlp:SessionIndex" minOccurs="0"
maxOccurs="unbounded"/>
                </sequence>
                <attribute name="Reason" type="string" use="optional"/>
                <attribute name="NotOnOrAfter" type="dateTime" use="optional"/>
            </extension>
        </complexContent>
    </complexType>
    <element name="SessionIndex" type="string"/>
    <element name="LogoutResponse" type="samlp:StatusResponseType"/>
    <element name="NameIDMappingRequest" type="samlp:NameIDMappingRequestType"/>
    <complexType name="NameIDMappingRequestType">
        <complexContent>
            <extension base="samlp:RequestAbstractType">
                <sequence>
                    <choice>
                        <element ref="saml:BaseID"/>
                        <element ref="saml:NameID"/>
                        <element ref="saml:EncryptedID"/>
                    </choice>
                    <element ref="samlp:NameIDPolicy"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NameIDMappingResponse"
type="samlp:NameIDMappingResponseType"/>
    <complexType name="NameIDMappingResponseType">
        <complexContent>
            <extension base="samlp:StatusResponseType">
                <choice>
                    <element ref="saml:NameID"/>
                    <element ref="saml:EncryptedID"/>
                </choice>
            </extension>
        </complexContent>
    </complexType>
</schema>
```

A.30 SAML Schema — X.500

下表列出 X.500 SAML Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<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>
```

A.31 SAML Schema — XACML

下表列出 XACML SAML Schema。

```
<?xml version="1.0" encoding="UTF-8"?>
<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>
```

附 录 一

安全与私密性考虑

必须系统性地解决安全与私密性，如社会工程攻击、政策问题、密钥管理和信任管理、安全实现以及其他因素等人的问题不在本附录的范围内。安全技术解决方案有一定消费，因此还要考虑到需求与政策选择以及法律和政策的监管要求。

本附录总结了一般的安全问题和方法以及特定的威胁和在安全操作下保持私密性时使用 SAML 断言、协议、绑定和协议子集的应对策略。本附录描述和分析了 SAML 的安全与私密特性，分为如下几个部分描述构建和实现基于 SAML 的系统：

- SAML 体系架构如何解决隐私性问题以及 SAML 以及 SAML 体系架构解决这些问题。
- 基于 SAML 的系统对威胁和安全风险问题来将是主体；
- SAML 体系架构解决安全风险问题，应如何解决；
- 不解决安全风险问题；
- 降低安全风险的建议对策。

I.1 私密性

SAML 包括针对被认证实体的属性和认证进行陈述的能力。具有非常多的共同情况，在这些情况中，这些陈述中携带的信念是一些通信一方或多方希望保持可接入到尽可能限制的实体。医学上或金融上的属性是这类情况的简单例子。许多国家和地区都在法律法规方面考虑到隐私问题，当部署一个基于 SAML 的系统时也要考虑到这个问题。当事人进行陈述、发布断言、传送断言、消费断言的一方时必须关注到潜在的隐私问题，并且应该在 SAML-aware 系统执行过程中尝试提出这个问题。

I.2 保密性

或许，确保可以进行 SAML 的交易中各方的隐私的最重要的特性是执行具有保密性保证的能力。换言之，断言中的信息是否能够由发布方传递到目的接收方，而不是其他接收方。

传递信息保密性在技术上是可行的。采用 SAML 的交易的各方应当分析它们在没有使它可以接入到任何其他方时，在交易过程中的每一个步骤，（从交易获得数据的任何后续使用）确保需要保密的信息实际上保持其保密性。

应注意简单地对断言内容进行遮盖并不能充分保护私密性。许多场景中对给定用户（或 IP 地址）正在接入到给定的业务的信息的有效性可能构成保护私密性不可缺少的（例如，用户接入医学测试设施用于断言在没有知道断言的内容足够破坏私密性）。这些问题的部分解决计划按下面部分描述对匿名交易由各种技术提供的。

I.3 假名与匿名

没有任何场景均适用的匿名的定义。许多定义适合于发送端和消息的简单场景，同时讨论“匿名”是指不能将给定的发送端与发送的信息联系起来，或者将一个消息返回给一个发送端。当那个定义适合于“one off”场景时，它忽略了可能基于行为而不是一个标识符超时的信息的聚合。

在 SAML 中，将匿名看做是“在一个集中是有帮助的”。由于授权中心的使用，该标志与 SAML 相关。即使一个主体是“匿名的”，那个主体也可以作为相关授权中心的域中的主体集的一个成员可标识。可以进行 SAML 的系统被限制到部分匿名最好，因为使用了授权中心，一个为其进行断言的实体已经可以作为在发布授权中心的关系中的实体池中的一个。

对匿名的限制可能比简单授权相关性还差，如取决于标识符如何被使用，由于假标识符的重用允许潜在的标识信任的增加/另外，可以使用 SAML 系统的用户也根据他们的行为使得匿名的破坏更糟。

除了合法的身份外，主体的任何标识符都可能被认为是“假名”。甚至使“密钥持有者”的标明可能被认为将动作（或一组动作）与一主体连接起来中假名的等价物。甚至，像“在 23:34 请求接入到对象 XYZ 的用户”可以作为假名的等价物。

那样，对“能伤害”来讲，将用户描述为一个标识符还是根据行为来描述没有什么不同，（例如，密钥的使用或动作的执行）。

以什么样的程度使用假名的特定等价物会不同。匿名给分类学从一直使用的个人假名（如昵称）开始，还有各种类型的角色假名（像防御的秘书）到“一次使用的”假名。

仅有一次性使用的假名可以给你匿名（在 SAML 中，将其认为是“一个集中的匿名”）。然后，给定的假名使用得越来越多，匿名的风险越大。换言之，假名的重复使用允许附加的潜在的标识与假名相关联的信息。随着时间的推移，这将导致可以标识与假名相关联的身份的增加。

起始站点授权中心（如以认证授权中心和属性授权中心）可以通过使用一次性使用的标识符或密钥来提供。一定的“部分匿名”（对于“密钥持有者的情景”）。由于主体有必要禁闭与授权中心相关的主体集。在使用集合属性时，该集可能被进一步减少（以进一步减少匿名）在起始站点的用户社团的那个进一步的子集。真正关心匿名的用户必须小心地假装或避免可能随着时间可用做“解匿名”的不正常的行为模式。

I.4 安全

下面几个小节讨论安全相关问题。

I.4.1 背景

基于计算机的系统间的通信存在的各种安全威胁，这些安全威胁具有某些相关联的风险。风险的特征取决于多种因素，包括通信的基本属性、通信系统的基本属性、通信媒介、通信环境、终端系统环境等因素。

SAML 的目的在于帮助开发者在安全域内或者域间在基于计算机系统的应用层面通信时建立安全关联。作为这种角色，SAML 传送认证数据，支持防止终端系统未经授权擅自使用的能力。SAML 就是用于保证通信安全。SAML 的威胁模型主要解决系统的安全问题。

I.4.2 范围

对 SAML 使用过程中系统全部的安全问题明显不在 SAML 的范围内。虽然本建议书不解决这些领域，在考虑系统的安全时总是要考虑这些领域。特别地，这些问题是重要的，但目前给出了 SAML 的范围：

- 初始化认证：SAML允许对已经完成的认证进行声明，但不包括这些动作的要求或规范。认证断言的使用者不应当盲目相信这些断言，除非他们了解这些断言是建立在什么基础之上的。对断言的信任绝不能超出断言方已经正确地得出断言的结论的信任。
- 信任模型：在很多情况下，SAML会话安全取决于下层的信任模型，该信任模型又是建立在密钥管理架构上的，（如PKI或者密钥）。例如通过XML签字进行安全保护的SOAP消息，只在密钥交换阶段的密钥是可信任时适用。未侦测的攻破密钥或者撤销的证书允许不违背安全模型。PKI的建立必须严格遵守安全模型，以保证上层协议的安全性。

安全协议的执行是维护系统安全所需要的，包括保护随机数或者伪随机数的生成、安全密钥存储等。

I.4.3 SAML威胁模型

SAML 威胁模型是以 IETF 制定的安全相关标准中互联网威胁模型为研究基础的。这里，我们假设两个或者多个 SAML 事务交互中的节点是未被攻破的，但是攻击者完全掌握通信信道。

另外，SAML 具有多方认证和授权的断言协议特点，必须要考虑到合法的 SAML 交互中的各方在哪里，谁在后续的交互中使用从前一次交互中收集到的信息。

下面描述几个可能受到攻击的场景：

— **联合攻击**：两个或者多个系统间联合展开的攻击，如：

  • 责任人和服务提供商之间的联合攻击；

  • 责任人和身份提供者之间的联合攻击；

  • 身份提供者和服务提供商之间的联合攻击；

  • 在两个或者多个责任人之间的联合攻击；

  • 在两个或者多个服务提供商之间的联合攻击；

  • 在两个或者多个身份提供者之间的联合攻击。

— **业务否认攻击**：防止经过授权后接入系统资源或者延迟系统操作和功能。

— **中间人攻击**：攻击者对会话进行窃听，并且修改通信数据，伪装为通信关联中的一个或者多个通信实体。

— **重放攻击**：攻击者对有效数据转换进行恶意的或者欺骗性重复。

— **会话拦截**：攻击者掌握前一次建立通信关联的控制进行窃听。

在所有的场景中，系统采用本地机制是否生成断言不在本标准的讨论范围内。这样像由在认证授权中心起始登录的细节引起的威胁也不在讨论的范围之内。如果授权中心签发一假的断言，那么由下游系统使用那个断言而引起的威胁明显地不在讨论范围内。

这样界定范围的直接是基于断言的系统安全仅仅在生成断言、修正数据和处理生成的断言时有用。当判断哪些发布者可以信任、哪些断言会作为输入认证或者授权判断、使用错误的但经过验证的发布的断言引发巨大的安全风险问题。在插入方和重放方之间的信任策略应当包括可靠性签字，在执行的时候应提供一套恰当的审计机制。

I.5 安全技术

下面各各节介绍安全技术及在 SAML 部署场景中适用的各种安全技术。

I.5.1 认证

认证是指事物交互中的一方确定另外一方身份的能力。认证可以是单向的，也可以是双向的。

— **有效会话（Active session）**：非永久认证由用于传送SAML消息的通信信道提供。这个认证可以是单向的，即由会话的初始方到接收方，也可以是双向的。根据使用的通信协议确定特定的方法。例如：使用TLS或者IPSec等安全协议，提供SMAL消息的发送方在TCP/IP环境中对目的方进行认证的能力。

— **消息层（Message-level）**：W3C XML 签字和OASIS WSS提供建立永久认证的方法，该方法与文档是紧耦合的。该方法不单独保证发送方的真实性。（的确，在许多情况下，包含中间媒介，明显不是这种状况）。允许永久确认包含一个唯一的可解析的实体，且该实体具有XML消息的特定子集的方法可以满足这个要求。

I.5.2 保密性

保密性是指消息的内容只能通过指定的接收方进行解析，而其他实体不能阅读消息体的内容。

— **在途**：通过使用TLS或者IPSec安全网络协议提供两点之间消息的传送安全。

— **消息层要求**：XML加密可以为XML文档提供有选择的加密。这个加密方法在XML消息中提供永久的、元素可选择的保密性。

I.5.3 数据完整性

数据完整性是确认接收到的消息没有被篡改，与发送方发送的消息版本一致。

— **传送**：像TLS或IP安全协议的安全网络协议的使用可以被配置来提供通过网络连接提供传送分组的完整性保护。将TLS或者IPSec配置为分组包通过网络连接传送时提供完整性保护。

— **消息层（Message-level）**：XML签字提供为消息完整性建立永久保证的方法，该方法与消息紧耦合。允许XML消息的稳定子集中一个永久的特定子集是满足这个要求。

I.5.4 基于密钥管理的节点

本附录着重介绍整个系统通过数字签字和加密等机制提供认证、数据完整性和保密性的能力。所有这些机制提供的安全都是局限在建立在密钥管理系统上的，局限性包括：

1） **接入到密钥**：假设基于密钥的系统用于认证、数据完整性和不可抵赖，保证安全接入私有的或者秘密密钥代表责任人对不相称的群体是不可用的。例如，根据Bob的私有密钥建立的数字签字仅仅证明Bob是唯一可以接入该密钥的人。通常，接入到密钥应当保留实体的最小设置，并且由通过短语或者其他方式进行保护。标准的安全防范应用。

2） **将身份与密钥绑定**：用于认证的基于密钥的系统必须具备一些身份与密钥可信任的绑定。验证文档的数字签字能够判断出文档签字以后是否被替换过，文档是否由特定的密钥进行签字。但是，这还不能够保证文档签字所使用的密钥就是特定的人在特定的时间内用的密钥。如果需要确认签字所使用的密钥就是发端用的密钥，还需要其他的验证手段进行证实。这种密钥到个人的绑定方式必须建立统一的解决方案，包含本地目录，用于存储标识符和密钥或者存储证书。证书的使用能够提供一种可扩展的将密钥和身份相关联的方法，但是需要机制配合管理证书的生命周期以及根据绑定的状态进行变更。

公开密钥架构（PKI）可以满足上述要求。一系列可信任的根认证授权中心（CA）确定某一个消费者的签字，回答“我该相信谁做身份到密钥绑定”。验证签字有效性第1步应判断问题中的签字是否就是由那个密钥签署的，第2步验证证书链，即密钥是否与绑定到正确的身份上且密钥仍可用。验证绑定的有效性需要保证绑定目前是有效的，证书都有一个生命周期，如果密钥在证书的生命周期里被破坏，那么密钥到身份绑定就失效。证书的关联可以在生命周期过期以前结束。一个合适的密钥管理系统需要有很好的健壮性但又不能过于复杂。签字有效性验证结束后，需要验证文件到密钥绑定的有效性，然后验证密钥到身份绑定的有效性。

I.5.5 TLS密码套件

本建议书在许多地方强烈建议采用 HTTP over SSL 3.0（见附录四）、HTTP over TLS 1.0 或者具备 HTTPS URL 方案的 URL 的使用。

除非指定，一般 SAML 绑定使用 SSL 3.0 和 TLS 1.0，服务器必须使用 X.509 v3 证书对客户端进行认证。客户端必须在证书内容的基础上建立服务器身份（典型地通过检查证书主体 DN 字段）。

可以配置 SSL/TLS 采用多种密码套件，并不是所有这些都是不提供足够的"最优方法"安全。SSL 密码套件结合了四种安全特性。数据在通过 SSL 连接传送之前，端点之间需要协商密码套件。端点之间建立一条可以保护通信的连接。与密码套有关的特征有：

SSL 定义了很多密钥交换算法。其中一些算法提供服务器认证，也支持匿名密钥交换机制。“RSA”认证的密钥交换算法是现在互操作性最好的算法。另一个重要的密钥交换算法是“DHE_DSS”密钥交换，不受相关专利执行的限制。

不管密钥交换算法是否能够从美国自由出来，都必须使用 512 比特短公开密钥用于密钥交换和 40 比特短对称密钥用于加密。这些长度的密钥均已经被成功攻击，因此不建议使用。

最快的加密算法选项是 RC4 流密码，密码块链条（CBC）模式支持 DES、DES40、3DES-EDE 和 AES。支持的其他模式见 TLS 文档。

空加密是在某些密码套件中是可选的。空加密就是不加密，使用于 SSL/TLS 仅进行认证和提供完整性保护的场景。空加密密码套件不提供保密性，因此不适用于要求保密性的环境。

文摘算法用于消息认证码。FCC 最近建议采用 SHA-256，IETF 也决定遵从 FCC 建议。

I.6 全面SAML安全考虑

下面各部分分析使用和执行 SAML 的安全风险，描述降低安全风险的对策。

I.6.1 SAML 断言

在 SAML 断言自身的层面，关于安全问题没什么可多说的 — 大部分安全问题是在请求/响应协议的通信过程中产生的，或是在试图通过一种邦定方法使用 SAML 的过程中产生的。当然，消费者总是希望能信赖 SAML 断言的有效间隔和出现在断言中的<OneTimeUse>元素。

不过，在断言层面还有个问题需要分析：断言一旦发布，即不受发布者的控制。这个事实有多个含义。例如，发布者无法对系统中断言的使用者持有断言的时间进行控制；发布者也无法对谁与断言的使用者共享断言信息进行控制。这些问题会比恶意攻击者通过搭接未加密（或加密不够严格）的线路而看到断言的内容更严重。

尽管在 SAML 建议书中对此类问题进行了充分讨论，但本建议书决没有放松对断言中应有的内容的审慎考虑。任何时候，发布者都应当考虑断言信息存储在远端站点的后果，如断言信息被直接滥用，或暴露给潜在的攻击者，或存储下来用于一些有创造性的欺骗用途。发布者还应当考虑断言中的信息可能与其他群体共享、甚至公开的安全问题。

I.6.2 SAML 协议

本节介绍 SAML 请求/响应协议自身设计的安全考虑，而不是来自使用某个特定协议绑定的威胁。

— **拒绝服务**

SAML 协议容易受到拒绝服务攻击。SAML 请求的处理需要较大开销，包括解析请求消息、数据库和断言存储查询、构造响应消息、对一个或者多个数字签字进行操作。攻击方生成请求消息的开销要比处理这些请求的开销小很多。

1） 要求客户端在较低层认证

要求客户端在SAML协议层的下层进行认证，例如SOAP over HTTP绑定，HTTP over TLS/SSL，要求在客户端侧进行认证，能够保证在根节点就具备可信任的认证授权，这样可以对DOS攻击进行追踪。

如果只使用认证对DOS攻击追踪，其本身并不阻止攻击的发生，但可以起到威慑作用。

如果认证具备一些进入控制系统，那么可以有效阻隔外部的DOS攻击。

客户端认证不管采用什么系统，都应当提供确认每一个请求发端的能力，不应当受到伪造。

2） 要求签字请求

规定还要求减少已做的工作之间的不对称的请求端和响应端的签字请求。响应端需要额外完成验证签字的工作在整个工作比例中只占很小的一部分，请求端计算数字签字的计算过程工作量非常大。缩小这种工作不对称性会降低DOS攻击相关的风险。

然而理论上攻击方可以签字一个消息，然后不停重放这个签字消息，绕开签字请求要求。这个漏洞可以通过要求使用包含时间戳的XML签字元素（如<ds:SignatureProperties>来避免。时间戳能够用于判断签字是否是最近一次的。在这种情况下，发布的签字在时间窗内被验证有效，需要应对更高的安全问题，重放拒绝服务攻击。

3） 限制进入互动

限制在非常低的层次向SAML业务发布请求，可以降低DOS攻击点风险。在这种场景下，只有来自有限集内的已知域的参与方生成的攻击，极大减少向潜在的恶意客户端和使用妥协机制的DOS攻击暴露。

有许多限制接入的方法，例如将SAML响应客户端放置在安全的企业网内，并且在路由层面使用准入规则。

I.7 SAML绑定安全考虑

SAML 请求响应协议在设计方面的安全考虑主要根据已使用的特定协议绑定。SAML 支持 SOAP 绑定、反向 SOAP 绑定（PAOS）、HTTP 重定向邦定、HTTP 重定向/ POST 绑定、HTTP 人造邦定和 SAML URI 绑定。

I.7.1 SAML SOAP绑定

SAML SOAP 绑定不要求认证也没有在途保密性或消息完整性要求，它开放给各种常见的攻击。下面从与 SOAP-over-HTTP 角度介绍一些安全考虑。

1） 窃听

威胁：如果没有在途保密性要求，窃听群可能截获包含请求消息的SOAP消息和包含响应消息的SOAP消息。这样会暴露请求消息和响应消息的具体内容，可能包括一个或者多个断言。这个问题在某些情况下会暴露请求端请求的断言类型，降低其安全性。

如果窃听者能够判断站点X经常用特定确认方法向站点Y请求认证断言，那么他也可以利用这些信息对站点X进行攻击。

类似的，对一系列的授权查询进行窃听能够建立一张在已知授权权威控制下的资源地图。
另外，暴露请求消息本身也会破坏私密性。例如，窃听一个查询请求及其响应消息，会暴露在查询站点活动的用户私密性信息，如医疗信息站点等。响应消息中的断言细节内容应当是保密的。如果响应消息中包含属性断言，这些属性表示信息是不能被事务交户参与方以外的实体使用的，那么窃听的安全风险就非常高。

应对措施：解决窃听攻击，应当提供某种形式的在途消息保密。对于SOAP消息，保密性可以在SOAP消息层提供，也可以在SOAP传输层提供。

在SOAP消息层增加在途保密性意味着构造SOAP消息时不考虑SOAP传输层可能有群接入的情况，这个方案和XML加密类似。本规范允许SOAP消息自身进行加密，可以降低窃听的风险。另外，部署方可以在SOAP传输层或者传输层的下层提供在途保密性。

如何提供保密性主要取决于选择哪种SOAP传送方式。HTTP over TLS/SSL是一种方式。其他传输方式需要其他的在途保密性技术，例如，SMTP传输会使用S/MIME方式。

在某些情况下，SOAP传输层的下层可以提供所需的在途保密机制。例如，如果采用IPsec隧道方式传送请求/响应交互消息，那么额外的在途保密性可以由隧道本身提供。

2） 重放

威胁：SOAP绑定层不易受重放攻击，重放攻击对各类协议子集有威胁。在SOAP绑定层重放攻击首要带来的问题就是利用重放进行拒绝服务攻击。

应对措施：通常，防止重放攻击最好的办法就是防止消息被捕获。一些提供在途保密性的传输层机制可以解决这个问题。例如，SAML的请求/响应会话通过在HTTP/TSL之上的SOAP消息传送，可以防止第三方截获消息。

潜在的重放攻击者不需要解析消息，只是重复发送该消息，XML加密机制不能提供对重放攻击的保护。如果攻击方能够截获发送给响应端的经请求端签字并加密的SAML请求，那么它可以在任何时间重放该请求，而不需要对消息进行解密。SAML请求包括请求发布的时间、可允许的目的地。

另外，请求消息的唯一密钥，即请求消息ID可以用于判断请求消息是否为重放的请求。重放攻击还会威胁“基于请求进行计费”模型。重放会导致某些用户账号生成大量的计费信息。

类似的，如果在系统中给一个客户端分配了固定号码，那么重放攻击能够耗尽用户资源，除非发布方注意跟踪唯一请求的唯一密钥。

3） 消息插入

威胁：伪造的请求或者相应消息插入到消息流里。伪造的响应消息返回给授权判断查询或者返回响应消息中包含伪造的属性，会导致属性查询方按照接收到的伪造响应进行错误的行动。

应对措施：插入请求消息的能力在SOAP绑定层不构成威胁。构成威胁的是插入伪造的响应会带来拒绝服务攻击。在返回伪造响应消息中更细小的攻击出现在SAML协议中，根据SOAP绑定定义，每一个SOAP响应消息必须包含一个单独的SAML协议响应，除非这个响应消息是伪造的。SAML协议提出两种机制来支持这个方法，第一，请求消息的响应要求使用InResponseTo属性，这就加大了攻击方进行攻击的难度，攻击方要先截获请求消息，然后再生成响应消息；第二，支持源认证，可以通过签字的SAML响应消息或者通过安全保护的传输连接如SSL/TLS两种方式实现。

4） 消息删除

威胁：消息删除攻击会阻止请求消息到达响应端，或者阻止响应消息到达请求端。

应对措施：SOAP绑定不处理这种威胁。通常，请求/响应相关消息可以降低这种攻击，例如在`StatusResponseType`中使用`InResponseTo`属性。

5） 消息修改

威胁：消息修改对SOAP绑定的双向都有威胁。

修改请求消息，替换请求消息的细节，会返回完全不同的响应消息，聪明的攻击者会根据返回的断言威胁系统安全。例如，替换<Attribute>元素中的请求属性列表会导致破坏或者拒绝响应端的请求消息。

修改请求消息，替换发布方请求，能够导致拒绝服务攻击或者响应消息的不正确路由。在SAML层下层出现的替换不在本建议书范围内规定。

修改响应消息，替换断言的详细内容能够导致大面积的破坏。替换认证消息或者授权消息的结果能够导致各种严重的安全问题。

应对措施：为了解决这些潜在的威胁，系统要保证在途消息完整性。SAML协议和SOAP绑定既不要求也不禁止部署保证在途消息完整性的系统，但鉴于系统巨大的威胁，还是高度建议采用这种系统。在SOAP绑定层，可以通过对请求消息和响应消息进行数字签字，如XML签字。

如果消息经过数字签字，那么接收方可以确保消息在传送过程中没有被替换，除非使用的密钥已经被破坏。

在途消息完整性的目标可以通过采用SOAP传输在较低层完成，或者可以由其他协议实现。SOAP over HTTP over TLS/SSL可以提供完整性保证。

单纯的加密可以保证消息不被修改，但不能保证攻击方使用其他的消息对其进行替换，因此不能提供完整性保证。

6） 中间人

威胁：SOAP绑定容易受中间人攻击，为了防止中间人攻击，需要采用一些双向认证的机制。

应对措施：双向认证系统允许通信的两个群看到的消息就是从对端发来的消息。

在SOAP绑定层，也可以通过同时对请求消息和响应消息进行数字签字的方式来防范威胁。这种方法不能避免位于中间请求端和响应端之间的窃听者进行双向转发，但是可以避免对消息修改而不知情的情况。

很多SOAP应用不使用会话，这种类型的认证授权方需要结合传输层的信息对请求消息的发送方和授权方的身份进行确认，防止中间人窃听的威胁。

另一种实现方式依靠SOAP传出层或者下层提供的双向认证机制。这种方式和SOAP over HTTP over TLS/SSL方式相比，需要对服务器和客户端侧都进行证书验证。

另外，断言的有效间隔返回功能可以衡量中间人攻击威胁程度。断言的有效窗口越短暂，被监听的危险性越低。

7） 采用SOAP over HTTP

由于SOAP绑定要求一致的应用支持HTTP over TLS/SSL，完成一系列不同的双向认证方法，如基本的服务器侧SSL和后向证书认证的服务器侧SSL，这些方法都可用于降低下层系统不能保证安全性且上述列出的攻击又十分重大的威胁。

但这并不意味着采用具备一些双向认证机制的HTTP over TLS是必须的。如果采用其他安全机制（如IPsec隧道方式等）在可接受的安全等级内可以免受威胁风险，则不要求具备证书的完全TLS机制。然而，在大部分SOAP over HTTP应用场景中，采用HTTP over TLS进行双向认证是比较合适的选择。

HTTP认证（IETF RFC 2617）介绍了使用基本的或者消息摘要认证机制时，在HTTP环境中可能遭受的攻击。

然而，传输层安全仅仅提供了“一跳”的保密性、数据完整性、认证。对于有中间传输媒介和多余一跳的通信模型，支持TLS/SSL的HTTP就不能提供充足的安全保障。

I.7.2 Web浏览器单点登录（SSO）介绍协议子集

源站点和目的站点对用户的认证不在本标准讨论范围内。关键的问题是源系统实体必须能够确定与之通信的已授权的客户端系统实体就是下一个步骤中需要交互的实体，可以通过在交互过程的初始化阶段使用 TLS 协议作为会话层的下层协议来保证。

I.7.2.1 SSO介绍 协议子集

1） 窃听

威胁：所有的web浏览器都存在窃听的可能。

应对措施：在有保密性要求的情况下，HTTP流量需要在传送层之上。例如，HTTP over TLS/SSL和IPSec协议就可以满足保密要求。

2） 窃取用户认证信息

威胁：源站点主要针对重用认证信息进行鉴别，例如，盗取认证信息，如密码形式等，都有可能让攻击者伪造主体。

应对措施：为了避免这个问题的发生，在两个主浏览器和源站点之间必须建立一道保密围墙。另外，必须在主浏览器和目的站点之间采取措施，在解析认证信息之前确保源站点的可信性。可以采用HTTP over TLS的方式解决这个问题。

3） 窃取承载层令牌

威胁：当认证断言包含断言承载层的认证协议标识符，窃取承载层令牌会带来攻击者伪造主体的问题。

应对措施：下面的几种方法降低了发生上述问题的可能性：

目的站点与主浏览器之间建立的连接使用保密围墙。

主或者目的站点确保源站点与主浏览器之间的连接是由保密围墙保护的。

目的站点验证主浏览器直接通过源站点进行重定向，直接由源站点进行认证。

源站点拒绝对相同断言ID的断言请求进行多次响应。

如果断言包含一个类型为**AudienceRestrictionTyp**的条件元素，这个条件元素标识一个特定域，则目的站点需要验证这个断言是否是属于特定域。

断言ID通过的目的站点和源站点之间建立的连接，是受保密围墙保护的。

断言ID通过的目的站点和源站点之间的通信，必须验证源站点的真实性和可信性。

4） 重放

重放攻击可能会带来业务否认和伪造找回信息两个问题。特定的协议绑定有特定的解决措施。

5） 消息插入

消息插入攻击见第I.7.1节。

6） 消息删除

威胁：在浏览器、SAML断言发布方和SAML断言使用方之间交互的任何一个步骤中删除消息都会造成交互失败，即拒绝某些服务，但不会暴露更多的信息。

应对措施：当不出现中间媒介时，采用完整性对传输信道进行完整性保护会造成消息删除的威胁。

7） 消息修改

威胁：更改数据流里消息是有可能的，会带来下述一些问题：

更改初始化请求消息，SAML发布方会拒绝该请求，或者创建一个伪资源，而不是请求消息要求的资源。

更改伪资源可能会导致SAML消费者拒绝服务。

在传送过程中更改断言会带来各种各样的不好的结果（如果未签字）或者会导致拒绝服务（如果签字了且消费者拒绝）。

应对措施：为了避免消息修改，应当在可以保证端到端消息完整性的系统中传递。

基于Web浏览器的用户数据，可以在具有数据完整性验证能力的HTTP over TLS/SSL上传送。

8） 中间人

威胁：中间人是对这套profile有很强破坏性的攻击。中间人能够重放请求、捕获返回的断言并将其篡改再传送。这样会导致起始方用户不能接入到所请求的资源，而中间人却可以通过截获的请求消息接入起始方所请求的资源。

应对措施：为防止出现这种威胁，需要采取几个应对措施。首先，采用一个可以提供强大的双向认证功能的系统，这样加大了中间人介入通信的难度。

然而，目前存在的中间人纯粹扮演的是双向端口转发的角色，进行信息窃取，获取返回的断言或者对其进行替换后再返回给请求端。如果部署一套加密系统，可以防止窃听。部署一套数据完整性系统可以防止中间人在断口转发时对消息进行修改。

对于本套profile，所有关于强大的双向会话认证、保密性、数据完整性的需求都可以通过使用HTTP over TLS/SSL（TLS/SSL层使用恰当的密码套件）进行传送，采用X509v3证书进行认证来解决。

9） 假扮无重认证

威胁：流氓用户企图假扮当前登录的合法主体，接入到受保护的资源。

一旦主体成功登录到身份认证提供商，后续从其他服务提供商发送来的<AuthnRequest>消息就不需要对主体进行重认证。主体必须进行认证，除非身份认证提供商可以判断<AuthnRequest>消息是否和主体身份关联的，同时，需要对主体进行的身份验证提供商会话进行有效地验证。

应对措施：身份验证提供商必须保持活动会话的状态信息，必须在发送响应消息之间验证<AuthnRequest>消息和活动会话之间的通信是否还有效，相应消息不需要对主体进行认证。身份验证提供商发布的cookie可以支持这个验证过程，但Liberty没有强制这个cookie-based的方法。

I.7.2.2 增强客户端和代理介绍

1） 中间人

威胁：截获SOAP消息的认证请求和响应，伪造主体。

伪造的系统实体能够作为中间人（MITM）在增强客户端和合法的服务提供商之间介入，同时扮演增强客户端和合法服务提供商两个角色，截取增强客户端和合法服务提供商之间的通信信息。首先，MITM截获服务提供商的认证请求消息，用其他URL代替它选择在PAOS报头块中的`responseConsumerServiceURL`值，然后再将篡改过的消息转发给增强客户端。通常，MITM会插入一个指向它自己的URL值。然后，增强客户端后续接收到身份认证提供商发送的响应消息，则会向MITM发送来的`responseConsumerServiceURL`发送响应消息，MITM伪装成合法的服务提供商侧的主体。

应对措施：身份验证提供商指定增强客户端的地址发送响应消息。PAOS报头中的`responseConsumerServiceURL`只用于发送错误响应。

2） 拒绝服务

威胁：改变`AuthRequest` SOAP请求是不能够被处理的，例如将PAOS报头块业务属性值改为未知值或者改变ECP报头块`Provider ID`或`IDPList`，会造成请求失败。

应对措施：通过使用SOAP消息安全或者SSL/TLS，对SOAP消息采用完整性保护。

I.7.2.3 身份认证提供商发现介绍协议子集

威胁：Cookie 中毒攻击，cookie 内部参数被修改，会出现发现伪造的身份认证提供商。

应对措施：使用共同主域机制能够限制这种威胁的发生。

I.7.2.4 单点退出介绍协议子集

威胁：被动攻击能够收集主体的名称标识符。

在初始化阶段，被动攻击方能够收集重定向信息中发布的`<LogoutRequest>`信息。这些数据构成威胁隐私数据曝光。

应对措施：所有的数据交换都应当建立在 SSL 或 TLS 安全传输层上。

威胁：未签字`<LogoutRequest>`消息

未签字`<LogoutRequest>` 可能被伪造的系统实体感染，因此拒绝服务。假设 `NameID` 能够被推导出，那么用户代理可能会传递一个伪造的`<LogoutRequest>`消息。

应对措施：对`<LogoutRequest>`消息签字。身份认证提供商能够验证缺少签字请求的主体的身份。

I.7.2.5 名称标识符管理介绍协议子集

威胁：允许系统实体对信息进行关联，否则会泄露身份信息，影响私密性。

应对措施：IDP 必须注意对相同的 principal，在和不同的服务提供商通信时要使用不同的名称标识符。IDP 应当对名称标识符进行加密，并返回给服务提供商，允许后续的交互使用不透明标识符。

I.7.2.6 属性介绍协议子集

与绑定相关的威胁是和属性相关联的。额外特定属性威胁目前未知。

附 录 二

MIME媒体类型应用/samlassertion+ xml的注册

本附录包含 SAML 应用/断言 MIME 媒体类型的注册。

MIME 媒体类型名称

— 应用

MIME 子类型名称

— Samlassertion+xml

需要的参数

— 没有。

选用的参数

— charset

— 与**IETF RFC 3923**中相同的应用/xml的charset参数

编码考虑

— 与IETF RFC 3923中相同的应用/xml的charset参数

安全考虑

本 samlassertion+xml 类对象不包含可执行的内容。然而，SAML 断言是基于 XML 的对象。同样的，由于它们是显式的安全对象，它们有在 IETF RFC 3923 中所给出的所有一般性安全考虑，同时具有附加的一些内容。例如，samlassertion+xml 类型的对象通常包含有可以标识或适合于自然人的数据，同时可以被用做为会话和接入控制决定的基础。

为能够追溯潜在的问题，samlassertion+xml 类型对象包含应由该发送端适当地签署的数据。任何这类签字必须由该数据的接收端核实-核实有效的签字以及是该发送端的签字。包含有 SAML 断言的 samlassertion+xml 类型对象也可以加密所有和部分断言。

另外，绑定的 SAML 协议子集和协议合适地规定安全信道的使用。

本标准在设计中合并了各种隐私保护技术。例如，不透明处理，在特定系统实体之间的特定交互，可以分配给主体。句柄可以是仅由特定方映射到较宽的关联标识符（例如，email 地址，账户标识符等）。

互操作性考虑

SAML 断言是显式版本化。信任方应确保它们观察断言版本信息并适当地动作。

出版的规范

本标准显式地规定了应用/ samlassertion+xml MIME 媒体类型的使用。然而，实际上可以使用 SAML 绑定传递非 SAML 断言（例如，SAMLv1 和/或 SAMLV1.1）。

使用该媒体类型的应用

潜在地，执行 SAML 的任何应用以及执行基于 SAML 的规范的那些应用。

附加信息

魔数 Magic number(s)

通常，与用于应用/xml 的相同。特别地，返回的对象的 XML 根单元将具有包含有如下一些内容的当限命名空间的名称：

- 本地名称：Assertion

- URI命名空间：一个由合适的特定版本SAML“核心”建议规定的特定版本SAML断言命名空间URI。

具有特定的 SAML，返回的对象的根单元可以是<saml:Assertion>或<saml:EncryptedAssertion>，此处“saml”代表映射到 SAML 断言命名空间 URI 的任何 XML 命名空间前缀：

urn: oasis:names:tc:SAML:2.0:assertion

文件扩展

无

Macintosh 文件类型码

无

用于联系进一步信息的人和 email 地址

代表 OASIS 安全业务技术委员会（SSTC）进行该注册。

希望的用途

通用

附 录 三

MIMI媒体类型应用/ samlmetadata+xml的注册

本附录规定与安全断言标识语言元数据一同使用的一 MIME 媒体类型—应用 samlmetadata+xml。

1） MIME媒体类型名称

应用

2） MIME子类型名称

samlmetadata+xml

3） 需要的参数

无。

4） 可选的参数

charset

与应用/ xml（见 IETF RFC 3023）的charset参数相同。

5） 编码考虑

与在IETF RFC 3023中用于应用/ xml的相同。

6） 安全考虑

samlmetadata+xml类对象不包含可执行的内容。然而，这些对象是基于XML的，因而，它们具有IETF RFC 3023第10节中给出的所有一般安全考虑。

为追溯潜在的问题，出版者可以签署samlmetadata+xml类对象。数据的接收端可以检查任何这样的签字-作为有效的签字并且作为出版者的签字。

7） 互操作考虑

SAML元数据显式地支持由标识的实体所支持的协议和版本。例如，若它们是可以通过URI无二意地可标识，任一标识提供者实体可以表示支持SAML v2.0和其他协议。该协议支持信息可以通过**RoleDescriptorType**的元数据对象的protocolSupportEnumeration属性承载。

8） 出版的建议

SAML元数据显式地规定应用/samlmetadata+xml MIME媒体类型的使用。

使用该媒体类型的应用：

潜在地，任何执行SAML v2.0的应用以及那些执行基于SAML的规范的应用。

9） 附加信息

1） 魔数

一般情况下，与在IETF RFC 3023中的`application/xml`相同。特别地,返回的对象的XML根单元将具有名称中采用如下内容的赋值命名空间：

— 本地名称：`EntityDescriptor`，或`AffiliationDescriptor`，或`EntitiesDescriptor`

— 命名空间URI：`urn:oasis:names:tc:SAML:2.0:metadata`（SAMLv2.0 元数据命名空间）

10）文件扩展

无。

11）Macintosh文件类型码

无。

12）为获得进一步信息联系人和email地址

代表OASIS 安全业务技术委员会（SSTC）进行注册。

13）打算的使用

公用。

附 录 四

SSL的使用

SAML 的某些实现可以另外或替代 TLS 1.0 支持 SSL 3.0 的使用。使用 SSL 3.0 的实现应确保实现的整体安全与在 TLS 中对密码的使用所做的限制一致。例如，使用密码组 TLS_RSA_WITH_3DES_EDE_CBC_SHA 的需求转化为 SSL_RSA_WITH_3DES_EDE_CBC_SHA 密码组的使用。FIPS 能够进行 SSL 的实现使用对应于 SSL SSL_RSA_WITH_3DES_EDE_CBC_SHA 密码组的 FIPS 密码组。

支持 TLS_RSA_WITH_3DES_EDE_CBC_SHA 密码组的 SAML 的 Web SSO 协议子集的实现将使用 SSL_RSA_WITH_3DES_EDE_CBC_SHA 密码组。

附 录 五

SAML Schema认证关联

本附录包含用于 SSL 证书（sslcert）的 SAML 认证关联方案。

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema targetNamespace="urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient"
  finalDefault="extension"
  blockDefault="substitution"
  version="2.0">

  <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">

    <xs:annotation>
      <xs:documentation>
        Class identifier: urn:oasis:names:tc:SAML:2.0:ac:classes:TLSClient
```

```
        Document identifier: saml-schema-authn-context-sslcert-2.0
        Location: http://docs.oasis-open.org/security/saml/v2.0/
        Revision history:
          V2.0 (March, 2005):
            New authentication_u99 context class schema for SAML V2.0.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType name="AuthnContextDeclarationBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnContextDeclarationBaseType">
          <xs:sequence>
            <xs:element ref="Identification" minOccurs="0"/>
            <xs:element ref="TechnicalProtection" minOccurs="0"/>
            <xs:element ref="OperationalProtection" minOccurs="0"/>
            <xs:element ref="AuthnMethod"/>
            <xs:element ref="GoverningAgreements" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
          <xs:attribute name="ID" type="xs:ID" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthnMethodBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthnMethodBaseType">
          <xs:sequence>
            <xs:element ref="PrincipalAuthenticationMechanism"/>
            <xs:element ref="Authenticator"/>
            <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PrincipalAuthenticationMechanismType">
      <xs:complexContent>
        <xs:restriction base="PrincipalAuthenticationMechanismType">
          <xs:sequence>
            <xs:element ref="RestrictedPassword"/>
          </xs:sequence>
          <xs:attribute name="preauth" type="xs:integer" use="optional"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorBaseType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorBaseType">
          <xs:sequence>
            <xs:element ref="DigSig"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="PublicKeyType">
      <xs:complexContent>
        <xs:restriction base="PublicKeyType">
          <xs:attribute name="keyValidation" type="xs:anyURI"
fixed="urn:oasis:names:tc:SAML:2.0:ac:classes:X509"/>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="AuthenticatorTransportProtocolType">
      <xs:complexContent>
        <xs:restriction base="AuthenticatorTransportProtocolType">
          <xs:sequence>
```

```
            <xs:choice>
              <xs:element ref="SSL"/>
              <xs:element ref="WTLS"/>
            </xs:choice>
            <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:restriction>
      </xs:complexContent>
    </xs:complexType>

  </xs:redefine>

</xs:schema>
```

附 录 六

认证关联类别XML 方案

本附录列出了用于单个一般断言的有效性的完全的关联类型 XML 方案和认证关联 XML 方案本身。类型方案不包含目标命名空间本身，而包含在附录五中。

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  version="2.0">

  <xs:annotation>
    <xs:documentation>
      Document identifier: saml-schema-authn-context-types-2.0
      Location: http://docs.oasis-open.org/security/saml/v2.0/
      Revision history:
          V2.0 (March, 2005):
          New core authentication context schema types for SAML V2.0.
    </xs:documentation>
  </xs:annotation>

  <xs:element name="AuthenticationContextDeclaration"
type="AuthnContextDeclarationBaseType">
    <xs:annotation>
      <xs:documentation>
        A particular assertion on an identity
        provider's part with respect to the authentication
        context associated with an authentication assertion.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="Identification" type="IdentificationType">
    <xs:annotation>
      <xs:documentation>
        Refers to those characteristics that describe the
        processes and mechanisms
        the Authentication Authority uses to initially create
        an association between a Principal
        and the identity (or name) by which the Principal will
        be known
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="PhysicalVerification">
    <xs:annotation>
      <xs:documentation>
```

```
      This element indicates that identification has been
      performed in a physical
      face-to-face meeting with the principal and not in an
      online manner.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:attribute name="credentialLevel">
      <xs:simpleType>
        <xs:restriction base="xs:NMTOKEN">
          <xs:enumeration value="primary"/>
          <xs:enumeration value="secondary"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>

<xs:element name="WrittenConsent" type="ExtensionOnlyType"/>

<xs:element name="TechnicalProtection" type="TechnicalProtectionBaseType">
  <xs:annotation>
    <xs:documentation>
      Refers to those characterstics that describe how the
      'secret' (the knowledge or possession
      of which allows the Principal to authenticate to the
      Authentication Authority) is kept secure
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SecretKeyProtection" type="SecretKeyProtectionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates the types and strengths of
      facilities
      of a UA used to protect a shared secret key from
      unauthorized access and/or use.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="PrivateKeyProtection" type="PrivateKeyProtectionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates the types and strengths of
      facilities
      of a UA used to protect a private key from
      unauthorized access and/or use.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="KeyActivation" type="KeyActivationType">
  <xs:annotation>
    <xs:documentation>The actions that must be performed
      before the private key can be used. </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="KeySharing" type="KeySharingType">
  <xs:annotation>
    <xs:documentation>Whether or not the private key is shared
      with the certificate authority.</xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="KeyStorage" type="KeyStorageType">
  <xs:annotation>
    <xs:documentation>
      In which medium is the key stored.
```

```
      memory - the key is stored in memory.
      smartcard - the key is stored in a smartcard.
      token - the key is stored in a hardware token.
      MobileDevice - the key is stored in a mobile device.
      MobileAuthCard - the key is stored in a mobile
      authentication card.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SubscriberLineNumber" type="ExtensionOnlyType"/>
<xs:element name="UserSuffix" type="ExtensionOnlyType"/>

<xs:element name="Password" type="PasswordType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that a password (or passphrase)
      has been used to
      authenticate the Principal to a remote system.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationPin" type="ActivationPinType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that a Pin (Personal
      Identification Number) has been used to authenticate the Principal
      to some local system in order to activate a key.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Token" type="TokenType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that a hardware or software
      token is used
      as a method of identifying the Principal.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="TimeSyncToken" type="TimeSyncTokenType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that a time synchronization
      token is used to identify the Principal. hardware -
      the time synchonization
      token has been implemented in hardware. software - the
      time synchronization
      token has been implemented in software. SeedLength -
      the length, in bits, of the
      random seed used in the time synchronization token.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Smartcard" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that a smartcard is used to
      identify the Principal.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="Length" type="LengthType">
  <xs:annotation>
    <xs:documentation>
      This element indicates the minimum and/or maximum
```

```
        ASCII length of the password which is enforced (by the UA or the
        IdP). In other words, this is the minimum and/or maximum number of
        ASCII characters required to represent a valid password.
        min - the minimum number of ASCII characters required
        in a valid password, as enforced by the UA or the IdP.
        max - the maximum number of ASCII characters required
        in a valid password, as enforced by the UA or the IdP.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ActivationLimit" type="ActivationLimitType">
    <xs:annotation>
      <xs:documentation>
        This element indicates the length of time for which an
        PIN-based authentication is valid.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="Generation">
    <xs:annotation>
      <xs:documentation>
        Indicates whether the password was chosen by the
        Principal or auto-supplied by the Authentication Authority.
        principalchosen - the Principal is allowed to choose
        the value of the password. This is true even if
        the initial password is chosen at random by the UA or
        the IdP and the Principal is then free to change
        the password.
        automatic - the password is chosen by the UA or the
        IdP to be cryptographically strong in some sense,
        or to satisfy certain password rules, and that the
        Principal is not free to change it or to choose a new password.
      </xs:documentation>
    </xs:annotation>

    <xs:complexType>
      <xs:attribute name="mechanism" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:NMTOKEN">
            <xs:enumeration value="principalchosen"/>
            <xs:enumeration value="automatic"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>

  <xs:element name="AuthnMethod" type="AuthnMethodBaseType">
    <xs:annotation>
      <xs:documentation>
        Refers to those characteristics that define the
        mechanisms by which the Principal authenticates to the
        Authentication Authority.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="PrincipalAuthenticationMechanism"
type="PrincipalAuthenticationMechanismType">
    <xs:annotation>
      <xs:documentation>
        The method that a Principal employs to perform
        authentication to local system components.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="Authenticator" type="AuthenticatorBaseType">
    <xs:annotation>
```

```
      <xs:documentation>
        The method applied to validate a principal's
        authentication across a network
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ComplexAuthenticator" type="ComplexAuthenticatorType">
    <xs:annotation>
      <xs:documentation>
        Supports Authenticators with nested combinations of
        additional complexity.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="PreviousSession" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        Indicates that the Principal has been strongly
        authenticated in a previous session during which the IdP has set a
        cookie in the UA. During the present session the Principal has only
        been authenticated by the UA returning the cookie to the IdP.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ResumeSession" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        Rather like PreviousSession but using stronger
        security. A secret that was established in a previous session with
        the Authentication Authority has been cached by the local system
        and is now re-used (e.g. a Master Secret is used to derive new
        session keys in TLS, WTLS).
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="ZeroKnowledge" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates that the Principal has been
        authenticated by a zero knowledge technique as specified in ISO/IEC
        9798-5.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="SharedSecretChallengeResponse"
type="SharedSecretChallengeResponseType"/>

  <xs:complexType name="SharedSecretChallengeResponseType">
    <xs:annotation>
      <xs:documentation>
        This element indicates that the Principal has been
        authenticated by a challenge-response protocol utilizing shared
        secret keys and symmetric cryptography.
      </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="method" type="xs:anyURI" use="optional"/>
  </xs:complexType>

  <xs:element name="DigSig" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates that the Principal has been
        authenticated by a mechanism which involves the Principal computing
```

```
        a digital signature over at least challenge data provided
        by the IdP.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AsymmetricDecryption" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        The local system has a private key but it is used
        in decryption mode, rather than signature mode. For example, the
        Authentication Authority generates a secret and encrypts it using
        the local system's public key: the local system then proves it has
        decrypted the secret.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AsymmetricKeyAgreement" type="PublicKeyType">
    <xs:annotation>
      <xs:documentation>
        The local system has a private key and uses it for
        shared secret key agreement with the Authentication Authority
        (e.g., via Diffie Helman).
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:complexType name="PublicKeyType">
    <xs:sequence>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="keyValidation" use="optional"/>
  </xs:complexType>

  <xs:element name="IPAddress" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates that the Principal has been
        authenticated through connection from a particular IP address.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="SharedSecretDynamicPlaintext" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        The local system and Authentication Authority
        share a secret key. The local system uses this to encrypt a
        randomised string to pass to the Authentication Authority.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="AuthenticatorTransportProtocol"
type="AuthenticatorTransportProtocolType">
    <xs:annotation>
      <xs:documentation>
        The protocol across which Authenticator information is
        transferred to an Authentication Authority verifier.
      </xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="HTTP" type="ExtensionOnlyType">
    <xs:annotation>
      <xs:documentation>
        This element indicates that the Authenticator has been
        transmitted using bare HTTP utilizing no additional security
        protocols.
      </xs:documentation>
```

```
  </xs:annotation>
</xs:element>

<xs:element name="IPSec" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Authenticator has been
      transmitted using a transport mechanism protected by an IPSEC session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="WTLS" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Authenticator has been
      transmitted using a transport mechanism protected by a WTLS session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="MobileNetworkNoEncryption" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Authenticator has been
      transmitted solely across a mobile network using no additional
      security mechanism.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="MobileNetworkRadioEncryption" type="ExtensionOnlyType"/>
<xs:element name="MobileNetworkEndToEndEncryption" type="ExtensionOnlyType"/>

<xs:element name="SSL" type="ExtensionOnlyType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Authenticator has been
      transmitted using a transport mechanism protected by an SSL or TLS
      session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="PSTN" type="ExtensionOnlyType"/>
<xs:element name="ISDN" type="ExtensionOnlyType"/>
<xs:element name="ADSL" type="ExtensionOnlyType"/>

<xs:element name="OperationalProtection" type="OperationalProtectionType">
  <xs:annotation>
    <xs:documentation>
      Refers to those characteristics that describe
      procedural security controls employed by the Authentication Authority.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="SecurityAudit" type="SecurityAuditType"/>
<xs:element name="SwitchAudit" type="ExtensionOnlyType"/>
<xs:element name="DeactivationCallCenter" type="ExtensionOnlyType"/>

<xs:element name="GoverningAgreements" type="GoverningAgreementsType">
  <xs:annotation>
    <xs:documentation>
      Provides a mechanism for linking to external (likely
      human readable) documents in which additional business agreements,
      (e.g., liability constraints, obligations, etc.) can be placed.
    </xs:documentation>
  </xs:annotation>
</xs:element>
```

```
<xs:element name="GoverningAgreementRef" type="GoverningAgreementRefType"/>

<xs:simpleType name="nymType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="anonymity"/>
    <xs:enumeration value="verinymity"/>
    <xs:enumeration value="pseudonymity"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="AuthnContextDeclarationBaseType">
  <xs:sequence>
    <xs:element ref="Identification" minOccurs="0"/>
    <xs:element ref="TechnicalProtection" minOccurs="0"/>
    <xs:element ref="OperationalProtection" minOccurs="0"/>
    <xs:element ref="AuthnMethod" minOccurs="0"/>
    <xs:element ref="GoverningAgreements" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="ID" type="xs:ID" use="optional"/>
</xs:complexType>

<xs:complexType name="IdentificationType">
  <xs:sequence>
    <xs:element ref="PhysicalVerification" minOccurs="0"/>
    <xs:element ref="WrittenConsent" minOccurs="0"/>
    <xs:element ref="GoverningAgreements" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="nym" type="nymType">
    <xs:annotation>
      <xs:documentation>
        This attribute indicates whether or not the
        Identification mechanisms allow the actions of the Principal to
        be linked to an actual end user.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>

<xs:complexType name="TechnicalProtectionBaseType">
  <xs:sequence>
    <xs:choice minOccurs="0">
      <xs:element ref="PrivateKeyProtection"/>
      <xs:element ref="SecretKeyProtection"/>
    </xs:choice>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="OperationalProtectionType">
  <xs:sequence>
    <xs:element ref="SecurityAudit" minOccurs="0"/>
    <xs:element ref="DeactivationCallCenter" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="AuthnMethodBaseType">
  <xs:sequence>
    <xs:element ref="PrincipalAuthenticationMechanism" minOccurs="0"/>
    <xs:element ref="Authenticator" minOccurs="0"/>
    <xs:element ref="AuthenticatorTransportProtocol" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="GoverningAgreementsType">
  <xs:sequence>
    <xs:element ref="GoverningAgreementRef" maxOccurs="unbounded"/>
  </xs:sequence>
```

```
</xs:complexType>

<xs:complexType name="GoverningAgreementRefType">
  <xs:attribute name="governingAgreementRef" type="xs:anyURI" use="required"/>
</xs:complexType>

<xs:complexType name="PrincipalAuthenticationMechanismType">
  <xs:sequence>
    <xs:element ref="Password" minOccurs="0"/>
    <xs:element ref="RestrictedPassword" minOccurs="0"/>
    <xs:element ref="Token" minOccurs="0"/>
    <xs:element ref="Smartcard" minOccurs="0"/>
    <xs:element ref="ActivationPin" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="preauth" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:group name="AuthenticatorChoiceGroup">
  <xs:choice>
    <xs:element ref="PreviousSession"/>
    <xs:element ref="ResumeSession"/>
    <xs:element ref="DigSig"/>
    <xs:element ref="Password"/>
    <xs:element ref="RestrictedPassword"/>
    <xs:element ref="ZeroKnowledge"/>
    <xs:element ref="SharedSecretChallengeResponse"/>
    <xs:element ref="SharedSecretDynamicPlaintext"/>
    <xs:element ref="IPAddress"/>
    <xs:element ref="AsymmetricDecryption"/>
    <xs:element ref="AsymmetricKeyAgreement"/>
    <xs:element ref="SubscriberLineNumber"/>
    <xs:element ref="UserSuffix"/>
    <xs:element ref="ComplexAuthenticator"/>
  </xs:choice>
</xs:group>

<xs:group name="AuthenticatorSequenceGroup">
  <xs:sequence>
    <xs:element ref="PreviousSession" minOccurs="0"/>
    <xs:element ref="ResumeSession" minOccurs="0"/>
    <xs:element ref="DigSig" minOccurs="0"/>
    <xs:element ref="Password" minOccurs="0"/>
    <xs:element ref="RestrictedPassword" minOccurs="0"/>
    <xs:element ref="ZeroKnowledge" minOccurs="0"/>
    <xs:element ref="SharedSecretChallengeResponse" minOccurs="0"/>
    <xs:element ref="SharedSecretDynamicPlaintext" minOccurs="0"/>
    <xs:element ref="IPAddress" minOccurs="0"/>
    <xs:element ref="AsymmetricDecryption" minOccurs="0"/>
    <xs:element ref="AsymmetricKeyAgreement" minOccurs="0"/>
    <xs:element ref="SubscriberLineNumber" minOccurs="0"/>
    <xs:element ref="UserSuffix" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="AuthenticatorBaseType">
  <xs:sequence>
    <xs:group ref="AuthenticatorChoiceGroup"/>
    <xs:group ref="AuthenticatorSequenceGroup"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ComplexAuthenticatorType">
  <xs:sequence>
    <xs:group ref="AuthenticatorChoiceGroup"/>
    <xs:group ref="AuthenticatorSequenceGroup"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="AuthenticatorTransportProtocolType">
```

```
    <xs:sequence>
      <xs:choice minOccurs="0">
        <xs:element ref="HTTP"/>
        <xs:element ref="SSL"/>
        <xs:element ref="MobileNetworkNoEncryption"/>
        <xs:element ref="MobileNetworkRadioEncryption"/>
        <xs:element ref="MobileNetworkEndToEndEncryption"/>
        <xs:element ref="WTLS"/>
        <xs:element ref="IPSec"/>
        <xs:element ref="PSTN"/>
        <xs:element ref="ISDN"/>
        <xs:element ref="ADSL"/>
      </xs:choice>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="KeyActivationType">
    <xs:sequence>
      <xs:element ref="ActivationPin" minOccurs="0"/>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="KeySharingType">
    <xs:attribute name="sharing" type="xs:boolean" use="required"/>
  </xs:complexType>

  <xs:complexType name="PrivateKeyProtectionType">
    <xs:sequence>
      <xs:element ref="KeyActivation" minOccurs="0"/>
      <xs:element ref="KeyStorage" minOccurs="0"/>
      <xs:element ref="KeySharing" minOccurs="0"/>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="PasswordType">
    <xs:sequence>
      <xs:element ref="Length" minOccurs="0"/>
      <xs:element ref="Alphabet" minOccurs="0"/>
      <xs:element ref="Generation" minOccurs="0"/>
      <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="ExternalVerification" type="xs:anyURI" use="optional"/>
  </xs:complexType>

  <xs:element name="RestrictedPassword" type="RestrictedPasswordType"/>

  <xs:complexType name="RestrictedPasswordType">
    <xs:complexContent>
      <xs:restriction base="PasswordType">
        <xs:sequence>
          <xs:element name="Length" type="RestrictedLengthType" minOccurs="1"/>
          <xs:element ref="Generation" minOccurs="0"/>
          <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="ExternalVerification" type="xs:anyURI" use="optional"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:complexType name="RestrictedLengthType">
    <xs:complexContent>
      <xs:restriction base="LengthType">
        <xs:attribute name="min" use="required">
          <xs:simpleType>
            <xs:restriction base="xs:integer">
              <xs:minInclusive value="3"/>
            </xs:restriction>
          </xs:simpleType>
```

```
      </xs:attribute>
      <xs:attribute name="max" type="xs:integer" use="optional"/>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="ActivationPinType">
  <xs:sequence>
    <xs:element ref="Length" minOccurs="0"/>
    <xs:element ref="Alphabet" minOccurs="0"/>
    <xs:element ref="Generation" minOccurs="0"/>
    <xs:element ref="ActivationLimit" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="Alphabet" type="AlphabetType"/>
<xs:complexType name="AlphabetType">
  <xs:attribute name="requiredChars" type="xs:string" use="required"/>
  <xs:attribute name="excludedChars" type="xs:string" use="optional"/>
  <xs:attribute name="case" type="xs:string" use="optional"/>
</xs:complexType>

<xs:complexType name="TokenType">
  <xs:sequence>
    <xs:element ref="TimeSyncToken"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="DeviceTypeType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="hardware"/>
    <xs:enumeration value="software"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="booleanType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="true"/>
    <xs:enumeration value="false"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="TimeSyncTokenType">
  <xs:attribute name="DeviceType" type="DeviceTypeType" use="required"/>
  <xs:attribute name="SeedLength" type="xs:integer" use="required"/>
  <xs:attribute name="DeviceInHand" type="booleanType" use="required"/>
</xs:complexType>

<xs:complexType name="ActivationLimitType">
  <xs:choice>
    <xs:element ref="ActivationLimitDuration"/>
    <xs:element ref="ActivationLimitUsages"/>
    <xs:element ref="ActivationLimitSession"/>
  </xs:choice>
</xs:complexType>

<xs:element name="ActivationLimitDuration" type="ActivationLimitDurationType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Key Activation Limit is
      defined as a specific duration of time.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationLimitUsages" type="ActivationLimitUsagesType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Key Activation Limit is
```

```
      defined as a number of usages.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="ActivationLimitSession" type="ActivationLimitSessionType">
  <xs:annotation>
    <xs:documentation>
      This element indicates that the Key Activation Limit is
      the session.
    </xs:documentation>
  </xs:annotation>
</xs:element>

<xs:complexType name="ActivationLimitDurationType">
  <xs:attribute name="duration" type="xs:duration" use="required"/>
</xs:complexType>

<xs:complexType name="ActivationLimitUsagesType">
  <xs:attribute name="number" type="xs:integer" use="required"/>
</xs:complexType>

<xs:complexType name="ActivationLimitSessionType"/>

<xs:complexType name="LengthType">
  <xs:attribute name="min" type="xs:integer" use="required"/>
  <xs:attribute name="max" type="xs:integer" use="optional"/>
</xs:complexType>

<xs:simpleType name="mediumType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="memory"/>
    <xs:enumeration value="smartcard"/>
    <xs:enumeration value="token"/>
    <xs:enumeration value="MobileDevice"/>
    <xs:enumeration value="MobileAuthCard"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="KeyStorageType">
  <xs:attribute name="medium" type="mediumType" use="required"/>
</xs:complexType>

<xs:complexType name="SecretKeyProtectionType">
  <xs:sequence>
    <xs:element ref="KeyActivation" minOccurs="0"/>
    <xs:element ref="KeyStorage" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="SecurityAuditType">
  <xs:sequence>
    <xs:element ref="SwitchAudit" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ExtensionOnlyType">
  <xs:sequence>
    <xs:element ref="Extension" minOccurs="0"  maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="Extension" type="ExtensionType"/>

<xs:complexType name="ExtensionType">
  <xs:sequence>
    <xs:any namespace="##other" processContents="lax" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

```
</xs:schema>


<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="urn:oasis:names:tc:SAML:2.0:ac"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:oasis:names:tc:SAML:2.0:ac"
  blockDefault="substitution"
  version="2.0">

  <xs:annotation>
    <xs:documentation>
      Document identifier: saml-schema-authn-context-2.0
      Location: http://docs.oasis-open.org/security/saml/v2.0/
      Revision history:
        V2.0 (March, 2005):
          New core authentication context schema for SAML V2.0.
          This is just an include of all types from the schema
          referred to in the include statement below.
    </xs:documentation>
  </xs:annotation>

  <xs:include schemaLocation="saml-schema-authn-context-types-2.0.xsd"/>

</xs:schema>
```

注 — 在附录四中示出了SSL的使用。

附 录 七

SAML DCE PAC 属性协议子集

本附录讨论用于分布式计算环境（DCE）、特权属性证书（PAC）（见开源 DCE）的 SAML 绑定协议子集。

VII.1 DCE PAC属性协议子集

DCE PAC 属性协议子集规定将 DCE PAC 信息表示为 SAML 属性名和值。它被用来标准化构成 DCE 责任人身份的原语信息和 SAML 属性集之间的映射。该协议子集构建在 11.4.9.3 中规定的 UUID 属性协议子集基础上。

1） 必需的信息

- **标识**：`urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE`（这也是在附件A中的相应DCE PAC属性协议子集中指派的目标命名空间）
- **联系方式**：security-services-comment@lists.oasis-open.org
- **说明**：见下文。
- **更新**：无。

2） PAC 描述

DCE PAC 是可以承载任意DCE注册属性的可扩展的结构，但是信息的核心集是针对责任人通用的并由成批的DCE身份组成：

- 责任人DCE“领域”或“单元”
- 责任人特定的标识符
- 责任人原语DCE本地组成员资格

- 原语DCE本地组成员资格的责任人集（多值）
- 不相关的组成员资格的责任人集（多值）

每一个这类属性的原语值是UUID。

3） SAML属性命名

本协议子集规定特定DCE新年系映射到SAML属性，然后规定实际的特定属性名而不是命名约定。

对于本协议子集所规定的所有属性，<Attribute>单元中的NameFormat XML属性必须具有值urn:oasis:names:tc:SAML:2.0:attrname-format:uri.。

为了人容易阅读的目的，对某些应用也可以有与URI一起携带可选的串名字的需要。可选的XML属性FriendlyName可以被用于该目的。

4） 属性名对照

当且仅当其NameXML 属性值在ITU-T X.667建议书的意义上相等，两个<Attribute>单元指同一个SAML属性。在该对照中FriendlyName属性没有意义。

5） 协议子集-特定XML属性

为与<Attribute>单元一同使用不规定附加的XML属性。

6） SAML属性值

该协议子集规定的每一个属性的原语值是UUID。在第11.4.9.3节中描述的URN句法用做表示这样的值。

然而，该协议子集允许与UUID值相关联的附加信息，该附加信息由友好的、人类可读的串、表示DCE单元或领域的附加UUID组成。该附加信息承载在XML命名空间urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE中规定的FriendlyName和领域XML属性中的<AttributeValue>单元中承载。尽管具有相同的基本目的，但这与第8节中规定的FriendlyName XML属性不同。

列出的下列方案示出在xsi:type[附件A]中如何使用协议子集-特定的XML属性和复杂类型：

```
<schema
targetNamespace="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE"
    xmlns:dce="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE"
    xmlns="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <annotation>
        <documentation>
            Document identifier: saml-schema-dce-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V2.0 (March, 2005):
                Custom schema for DCE attribute profile, first published in
SAML 2.0.
        </documentation>
    </annotation>
    <complexType name="DCEValueType">
        <simpleContent>
            <extension base="anyURI">
                <attribute ref="dce:Realm" use="optional"/>
                <attribute ref="dce:FriendlyName" use="optional"/>
            </extension>
        </simpleContent>
    </complexType>
    <attribute name="Realm" type="anyURI"/>
    <attribute name="FriendlyName" type="string"/>
</schema>
```

7） **属性定义**

下面是本协议子集中规定的SAML属性集。在每一种情况下，xsi:type XML属性可以包含在<AttributeValue>单元中，但是必须具有**dce:DCEValueType**值，此处dce前缀是任意的并切必须限定在XML命名空间urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE。这样的应用将需要确认属性消费者包括本协议子集规定的扩展方案。

a） 领域

该单值属性表示SAML断言主体的DCE领域或单元

Name: urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:realm

该单个<AttributeValue>单元包含鉴别SAML断言主体的DCE领域/单元的URN格式中的UUID以及包含领域串名的可选的协议子集特定的FriendlyName XML属性。

b） 责任人

该单值属性代表SAML断言主体的DCE责任人身份

Name: urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:principal

该单<AttributeValue>单元包含鉴别SAML断言主体DCE责任人身份以URN格式表示的UUID以及包含责任人串名的可选的特定协议子集FriendlyName XML属性。

可以包含协议子集特定的领域XML属性，同时必须包含鉴别SAML断言主体DCE领域/单元以URN格式表示的UUID。

c） 原语组

该单值属性表示SAML断言主体原语DCE组成员资格。

Name:

urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:primary-group

该单<AttributeValue>单元包含鉴别SAML断言主体原语DCE组以URN格式表示的UUID，以及包含组串名的可选的特定协议子集FriendlyName XML属性。

可以包含协议子集特定的领域XML属性，同时必须包含鉴别SAML断言主体DCE领域/单元以URN格式表示的UUID。

d） 组

该多值属性表示SAML断言主体DCE本地组成员资格。

Name: urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:groups

每一个<AttributeValue>单元包含鉴别SAML断言主体的DCE组成员资格以URN格式表示的UUID，以及包含组串名的可选的特定协议子集FriendlyName XML属性。

可以包含协议子集特定的领域XML属性，同时必须包含鉴别SAML断言主体DCE领域/单元以URN格式表示的UUID。

e） 不相关组

该多值属性表示SAML断言主体DCE不相关组成员资格。

Name:

urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:foreign-groups

每一个<AttributeValue>单元包含鉴别SAML断言主体的DCE不相关组成员资格以URN格式表示的UUID，以及包含组串名的可选的特定协议子集FriendlyName XML属性。

可以包含协议子集特定的领域XML属性，同时必须包含鉴别不相关组的DCE领域/单元以URN格式表示的UUID。

VII.2 SAML Schema dce

这是用于分布式计算环境（DCE）的 SAML 认证关联方案。

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<schema targetNamespace="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE"
    xmlns:dce="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE"
    xmlns="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="unqualified"
    attributeFormDefault="unqualified"
    blockDefault="substitution"
    version="2.0">
    <annotation>
        <documentation>
            Document identifier: saml-schema-dce-2.0
            Location: http://docs.oasis-open.org/security/saml/v2.0/
            Revision history:
            V2.0 (March, 2005):
                Custom schema for DCE attribute profile, first published in SAML 2.0.
        </documentation>
    </annotation>
    <complexType name="DCEValueType">
        <simpleContent>
            <extension base="anyURI">
                <attribute ref="dce:Realm" use="optional"/>
                <attribute ref="dce:FriendlyName" use="optional"/>
            </extension>
        </simpleContent>
    </complexType>
    <attribute name="Realm" type="anyURI"/>
    <attribute name="FriendlyName" type="string"/>
</schema>
```

VII.3 例子

下面是将 PAC 数据转换到属于在领域"example.com"中命名为"jdoe"的 DCE 责任人的 SAML 属性的一个例子，领域"example.com"是"cubicle-dwellers"和"underpaid"本地组和"engineers"不相关组的一个成员。

```
<saml:Assertion
xmlns:dce="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE" ...>
  <saml:Issuer>...</saml:Issuer>
  <saml:Subject>...</saml:Subject>
  <saml:AttributeStatement>
  <saml:Attribute
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
       Name="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:realm">
    <saml:AttributeValue xsi:type="dce:DCEValueType"
dce:FriendlyName="example.com">
    urn:uuid:003c6cc1-9ff8-10f9-990f-004005b13a2b
    </saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
       Name="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:principal">
    <saml:AttributeValue xsi:type="dce:DCEValueType"
dce:FriendlyName="jdoe">
    urn:uuid:00305ed1-a1bd-10f9-a2d0-004005b13a2b
    </saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
       Name="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:primary-gro
up">
    <saml:AttributeValue xsi:type="dce:DCEValueType"
       dce:FriendlyName="cubicle-dwellers">
    urn:uuid:008c6181-a288-10f9-b6d6-004005b13a2b
    </saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
       Name="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:groups">
    <saml:AttributeValue xsi:type="dce:DCEValueType"
       dce:FriendlyName="cubicle-dwellers">
    urn:uuid:008c6181-a288-10f9-b6d6-004005b13a2b
```

```
    </saml:AttributeValue>
    <saml:AttributeValue xsi:type="dce:DCEValueType"
dce:FriendlyName="underpaid">
    urn:uuid:006a5a91-a2b7-10f9-824d-004005b13a2b
    </saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute
NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
       Name="urn:oasis:names:tc:SAML:2.0:profiles:attribute:DCE:foreign-gro
ups">
    <saml:AttributeValue xsi:type="dce:DCEValueType"
dce:FriendlyName="engineers"
       dce:Realm="urn:uuid:00583221-a35f-10f9-8b6e-004005b13a2b">
    urn:uuid:00099cf1-a355-10f9-9e95-004005b13a2b
    </saml:AttributeValue>
  </saml:Attribute>
  </saml:AttributeStatement>
</saml:Assertion>
```

附 录 八

SAML的OASIS澄清

本附录另外地回顾在 OASIS 中针对 SAML v2.0 所做的工作。OASIS SAML 组决定作为单独的文档出版这些澄清注释（见 OASIS: 2006 PE）。这些澄清是非标准的并且不包含在 OASIS SAML 的版本 2.0。在本建议书中，这些回顾在本附录中列出以确保 SAML 实施者意识到在作为 OASIS 标准出版 OASIS SAML V2.0 的出版后发生的讨论。

VIII.1 潜在的勘误表：PE 14

描述：Allowcreate 需要更多的清晰的定义。

本建议书中适用性：

参考第 8.2.4.1 和第 8.2.6 节中合适的注释。此外，在第 8.2.6.3 节中下面提供小节的第二段的澄清。

若在请求中包括<Terminate>单元，请求提供者指示（以服务提供商的情况为例）不再从身份提供者接受断言或（在身份提供者的情况下）不再给服务提供商签发有关责任人的断言。

若接收提供者维护与命名标识符相关的状态，像标识符本身的值（在对方法标识符的情况下）、SPProvidedID 值。发送端同意的标识符的创建/使用等，然后，接收端能够执行根据命名标识符表示的关系已经被终止进行维护。

接收端代表有关责任人（例如，后来的<AuthnRequest>）执行的后续操作应当以与没有任何先前状态一致的方式执行。

终止是为在第 8.2.4 节中的认证请求协议中使用 AllowCreate 属性触发任何状态管理行为的潜在的清除步骤。不利用该属性的发展是希望避免使用<Terminate>单元或将其作为纯建议方式处理。

注意在大多数情况下（值得注意的例外是围绕 SPProvidedIDattribute 的规则）没有对标识符提供者或与回归状态生成和使用相关的服务提供商没有需求。因而，在<Terminate>单元是接收到的 450 时，不要求显式的行为。然而，若有关标识符的使用回归状态出现（如好像附上 SPProvidedID 属性），<Terminate>单元提供该状态应被删除的明确指示（若在某种形式标识为废除了）。

VIII.2 潜在的勘误表：PE 26

描述：SSO 协议子集需要澄清。

在本建议书中的适用性：下列条款澄清如下：

11.4.1.4.2 <Response>使用

若身份提供者希望返回差错，它必须在<Response>消息中不包括任何断言。否则，若请求成功（或若响应与请求不相关），<Response>单元必须符合下列内容：

- 若响应是无符号的，<Issuer>单元可以被省略，但若出现（或如响应是有符号的），它必须包含包含签发身份提供者唯一标识符，Format属性必须被省略或具有urn:oasis:names:tc:SAML:2.0:nameid-format:entity的值。
- 它必须至少一个<Assertion>。每一个断言的<Issuer>单元必须包含相应的身份提供者的唯一身份；Format属性必须省略或者具有urn:oasis:names:tc:SAML:2.0:nameid-format:entity的值。注意该协议子集假定单一相应身份提供者，同时响应中所有的断言必须由相同的实体签发。
- 若包含多个断言，那么每一个断言的<Subject>单元必须涉及相同的责任人，允许<Subject>单元的内容来区分（例如，使用不同的<NameID>或替换的<SubjectConfirmation>单元）。
- 为消费，使用该协议子集签发的任何断言必须包含一个具有至少一个包含<Subject>单元urn:oasis:names:tc:SAML:2.0:cm:bearer的方法的一个<SubjectConfirmation>单元。这样一个断言被命名为承载断言。承载断言可以包含附加的<SubjectConfirmation>单元。
- 也可以包括没有承载<SubjectConfirmation>的断言；附加的断言或<SubjectConfirmation>单元的处理不在本协议子集的范围内。
- 至少一个承载<SubjectConfirmation>单元必须包含其自身包含一个Recipient属性的<SubjectConfirmationData>单元，Recipient属性包含服务提供商的断言消费者业务URL以及在断言可以被投递期间限制窗口的NotOnOrAfter属性。它也可以包含一个限制客户端地址的地址属性，断言可以从该地址投递。它不必包含NotBefore属性。若包含消息在到<AuthnRequest>的响应中，那么InResponseTo属性必须匹配请求端的ID。
- 一个或多个承载断言集必须包含至少一个将责任人的认证反映到身份提供者的<AuthnStatement>。可以包括多个<AuthnStatement>单元，但是本协议子集不规定多个陈述的语义。
- 若身份提供者支持第11.4.1.4.5节中规定的Single Logout协议子集，任何认证陈述必须包括SessionIndex属性以允许服务提供商的按每个会话退出请求。
- 在承载断言中可以包括在身份提供者的判断力的其他陈述。特别地，可以包括<AttributeStatement>单元。<AuthnRequest>可以包含AttributeConsumingServiceIndex XML属性，该属性指704信息中有关第9节中希望的或需要的属性。该身份提供者可以忽略该一点，或在判断力上发送其他属性。
- 每一个承载断言必须包含包括作为<Audience>的服务提供商的唯一标识符的<AudienceRestriction>。
- 根据服务提供商的请求，可以包括其他条件（和其他<Audience>单元）或在身份提供者的判断力。（当然，服务提供商必须理解并接受所有这类条件以便于断言可认为的有效。）
- 若有，身份提供者没有责任遵守<AuthnRequest>中的请求的<Conditions>集。

11.4.1.4.3 <Response>消息处理规则

无论使用 SAML 绑定，服务提供商必须做如下工作：

- 核实出现在断言或响应的任何签字。
- 核实承载<SubjectConfirmationData>中的Recipient属性匹配<Response>或人造物被投递到的断言消费者业务URL。
- 核实承载<SubjectConfirmationData>的NotOnOrAfter属性没有通过，主体到提供者之间可允许的时钟偏差。
- 核实在承载<SubjectConfirmationData>中的InResponseTo属性等于其初始<AuthnRequest>消息，除非该响应不是主动提供的，在该情况下属性必须不出现。
- 核实依赖的任何断言在其他方面是有效的。注意当多个承载<SubjectConfirmation>单元可以出现，根据本协议子集进行的单一这样的单元的评估足够证实一个断言。然而，若出现不止一个断言，每个断言必须独立评估。
- 若承载<SubjectConfirmationData>包括一个地址属性，服务提供商可以检查用户代理的客户端地址。
- 无效或其主体证实需要不能被满足的任何断言应被废弃并不应用做该责任人的安全关联。
- 若用于为该责任人建立一个安全关联的<AuthnStatement>包含SessionNotOnOrAfter属性，一旦该时间到达，该安全关联应被废弃，除非服务提供商通过重复该协议子集的使用重新建立该责任人的身份。注意若出现多个<AuthnStatement>单元，应遵守最接近现在时间的SessionNotOnOrAfter值。

11.4.1.4.4 POST特定处理规则

若使用 HTTP POST 绑定来投递<Response>，每一个断言必须采用数字签字被保护。可以通过签署每一个单个的<Assertion>单元或通过签署<Response>单元来完成。

该服务提供商必须确保承载断言，不通过在基于<SubjectConfirmationData>中的 NotOnOrAfter 属性考虑为有效的断言的时间长度内维护使用的 ID 值，被重放。

参考资料

— **FIPS-197(2001)**, *Advanced Encryption Standard* (AES).
— **IETF RFC 1738**, *Uniform Resource Locators* (URL), 1994.
— **IETF RFC 2256**, *A Summary of the X.500(96) User Schema for use with LDAPv3, 1997*.
— **IETF RFC 2279**, *UTF-8, a transformation format of ISO 10646, 1998*.
— **IETF RFC 2743**, *Generic Security Service Application Program Interface Version 2, Update 1, 2000*.
— **DCE,** *Distributed Computing Environment (DCE)*, Open Source. See http://www.opengroup.org/dce.
— **OASIS Authentication Context 2.0**, *Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0*, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Bindings 1**, *Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML)*, November, 5, 2002, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Bindings 1.1**, *Bindings and Profiles for the OASIS Security AssertionMarkup Language (SAML)*, September, 22, 2003, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Bindings 2.0**, *Bindings for the OASIS Security Assertion Markup Language (SAML)* V2. 0, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Conformance 2.0**, *Conformance Requirements for the OASIS Security Assertion Markup Language (SAML)* V2.00, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Glossary 2.0**, *Glossary for the OASIS Security Assertion Markup Language (SAML)* V2.0, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Metadata 2.0**, *Metadata for the OASIS Security Assertion Markup Language (SAML)* V2. 0, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Errata Document 24**, *Revision 24 draft of the non-normative SAML V2.0 Errata document*, February 27, 2006, http://www.oasis-open.org/committees/download.php/16935/sstc-saml-errata-2.0-draft-24.pdf.
— **OASIS Protocol 1.0**, *Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML)*, November, 5, 2002, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Protocol 1.1**, *Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML)*, September, 22, 2003, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Protocol 2.0**, *Profiles for the OASIS Security Assertion Markup Language (SAML)* V2. 0, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS SAML 1.0,** *Security AssertionMarkup Language (SAML) Version 1.0 Specification Set*, November, 5, 2002, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS SAML 1.1**, *Security AssertionMarkup Language (SAML) Version 1.1 Specification Set*, September, 22, 2003, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Security 1**, *Security Considerations for the OASIS Security Assertion Markup Language (SAML)*, November, 5, 2002, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Security 1.1**, *Security Considerations for the OASIS Security Assertion Markup Language (SAML)*, September, 22, 2003, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS Security 2.0**, *Security and Privacy Considerations for the OASIS Security Assertion Markup Language (SAML)* V2. 0, March 15, 2005, http://www.oasis-open.org/apps/org/workgroup/security/.
— **OASIS XACML1-1**, *eXtensible Access Control Markup Language (XACML) V1.1*, 24 July 2003, http://www.oasis-open.org/apps/org/workgroup/xacml/.
— **OASIS XACML1-1**, *eXtensible Access Control Markup Language (XACML) V1.0*, 18 February 2003, http://www.oasis-open.org/apps/org/workgroup/xacml/.
— **OASIS XACML 2.0**, *eXtensible Access Control Markup Language (XACML) V2.0*, 1 February 2005, http://www.oasis-open.org/apps/org/workgroup/xacml/.
— **SSL3**, *The SSL Protocol Version 3.0*. See http://wp.netscape.com/eng/ssl3/draft302.txt
— **W3C Character Model(2004)**, Working draft, 27 October 2005, *Character Model for the World Wide Web 1.0: Normalization*.

ITU-T 系列建议书

A系列	ITU-T工作的组织
D系列	一般资费原则
E系列	综合网络运行、电话业务、业务运行和人为因素
F系列	非话电信业务
G系列	传输系统和媒质、数字系统和网络
H系列	视听及多媒体系统
I系列	综合业务数字网
J系列	有线网络和电视、声音节目及其他多媒体信号的传输
K系列	干扰的防护
L系列	电缆和外部设备其他组件的结构、安装和保护
M系列	电信管理，包括TMN和网络维护
N系列	维护：国际声音节目和电视传输电路
O系列	测量设备的技术规范
P系列	电话传输质量、电话设施及本地线路网络
Q系列	交换和信令
R系列	电报传输
S系列	电报业务终端设备
T系列	远程信息处理业务的终端设备
U系列	电报交换
V系列	电话网上的数据通信
X系列	**数据网、开放系统通信和安全性**
Y系列	全球信息基础设施、互联网协议问题和下一代网络
Z系列	电信系统使用的语言和一般性软件情况

瑞士印刷
2007年，日内瓦