OASIS Cross-Enterprise Security and Privacy Authorization (XSPA) – WS-Trust Healthcare Profile

Working draft 20 August, 2008

Document identifier:

xspa-ws-trust-profile-01

Location:

Editor:

Brett Burley, Department of Veterans Affairs (Near Infinity Corporation)

Duane DeCouteau, Department of Veterans Affairs (Edmond Scientific Company)

Mike Davis, Department of Veterans Affairs

David Staggs, Department of Veterans Affairs (SAIC)

Contributors:

Abstract:

This document describes how WS-Trust is leveraged by cross-enterprise security and privacy authorization (XSPA) to satisfy requirements pertaining to information-centric security within the healthcare community.

Table of contents

1. Introduction

XSPA encompasses the mechanisms to authenticate and administer, and enforce authorization policies controlling access to protected information residing within or across enterprise boundaries. The policies being administered and enforced relate to security, privacy and consent directives. In general, and with respect to this profile, WS-Trust works in concert with additional, supporting, lower-layer standards including WS-Security, WS-Policy and SAML to provide the overarching XSPA specification.

[XACML] is well suited for, and may be used to provide policy administration and enforcement within XSPA, leveraging a WS-based infrastructure where appropriate. However, this profile does not include the use of XACML within XSPA. XSPA does not mandate the use of XACML.

This document provides an overview of the major WS components of the XSPA profile. The profile then establishes how these components may be used to implement cross-enterprise access control requirements relevant to the healthcare community.

This profile does not address security required to protect message transactions such as digital signatures and encryption, but instead discusses how shared messages can be used to negotiate the necessary claims to access a protect resource.

It is assumed that the reader is somewhat familiar with the WS standards extended by WS-Trust.

1.1. Document Roadmap

- Presentation of requirements, non-goals and terminology
- Review of components that are encompassed by this XSPA profile
- Details of XSPA implementation within the healthcare community

1.2. Requirements and Non-goals

1.2.1. Requirements

Achieve cross-enterprise authentication and authorization of entities (e.g. user or server) within the healthcare community. This will be accomplished through XSPA by leveraging and extending existing and candidate OASIS standards.

1.2.2. Non-Goals

The following topics are outside the scope of this document:

- The use of XACML as means for creating rules and policy sets within or across security domains.

2. Terminology

The following definitions establish the terminology and usage in this profile:

Access Control System (ACS) –

Attribute Service - An attribute service is a Web service that maintains information (attributes) about principals within a trust realm or federation. The term principal, in this context, can be applied to any system entity, not just a person.

Claim – A claim is a declaration made by an entity (e.g. name, identity, key, group, privilege, capability, attribute, etc).

Metadata – Any data that describes characteristics of a subject. For example, federation metadata describes attributes used in the federation process such as those used to identify – and either locate or determine the relationship to – a particular Identity Provider, Security Token Service or Relying Party service.

Policy – A policy is a collection of policy alternatives.

Purpose of Use

Realm or Domain – A realm or domain represents a single unit of security administration or trust.

Relying Party – A Web application or service that consumes Security Tokens issued by a Security Token Service.

Security Token – A security token represents a collection of claims.

Security Token Service – A security token service (STS) is a Web service that issues security tokens (see [WS-Security]). That is, it makes assertions based on evidence that it trusts, to whoever trusts it (or to specific recipients). To communicate trust, a service requires proof, such as a signature to prove knowledge of a security token or set of security tokens. A service itself can generate tokens or it can rely on a separate STS to issue a security token with its own trust statement (note that for some security token formats this can just be a re-issuance or co-signature). This forms the basis of trust brokering.

Trust – Trust is the characteristic that one entity is willing to rely upon a second entity to execute a set of actions and/or to make set of assertions about a set of subjects and/or scopes.

3. WS-Trust Overview

This profile specifies the use of WS-Trust, an extension of WS-Security, as a token-type agnostic means for requesting, issuing, renewing, and validating security assertions. While the WS-Trust specification completely describes these activities, a brief overview is provided here describing the interactions between a web service requestor, security token service (STS) and web service provider.

The core component of WS-Trust is the STS. The authentication and authorization-related services provided by the STS are conducted on the frontline of the multi-layered approach of this profiles strategy for securing web services.

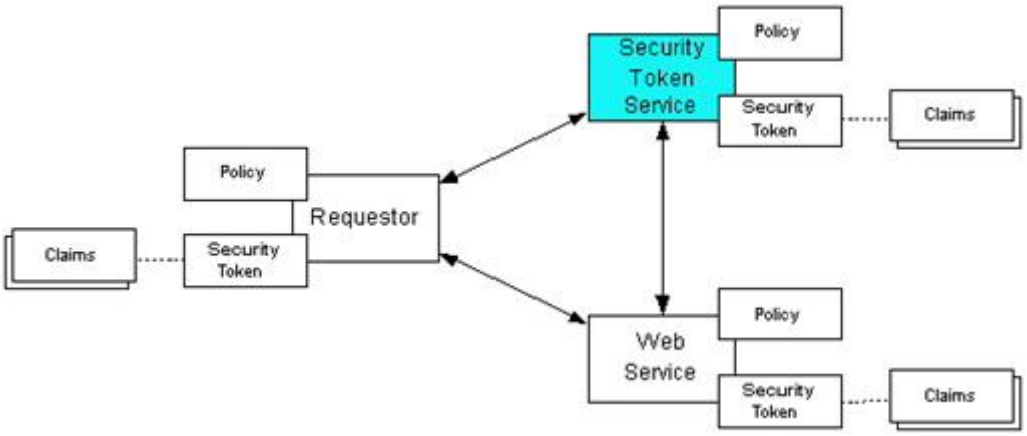

Figure 1 WS-Trust Model [WS-Trust]

In Figure 1 the STS acts as a brokering agent by issuing tokens, in the form of a response to the requestor. The requestor then submits the token to the web service. The web service may subsequently contact the STS to ensure the validity of the token and its claims (e.g. none of the claims have been revoked).

The most fundamental concept of WS-Trust is that requestors and consumers of tokens trust the issuer of that token, the STS.

4. XSPA WS-Trust Implementation

The objective of XSPA and this profile is to achieve cross-enterprise authorization of entities within the healthcare community by providing common semantics and vocabularies for interoperable course and fine-grained access control.

The following sections present a detailed look at the interaction between parties operating within the security framework of this profile, the elements of WS-Trust leveraged by XSPA, as well as a use case demonstrating the access control capabilities of XSPA.

4.1. Interactions between Parties

The XSPA WS-Trust model facilitates course and fine-grained access control, relieving the service provider from making access control decisions. The service provider is left with only having to enforce the decision determined by an access control service (ACS).

4.1.1. Assumptions

- The enterprise identity related to the healthcare system user has been established prior to issuing the request for a security token
- No user or application is allowed direct access to patient information services
- All security policies related to healthcare system end point have been established and put in place using mechanisms outside the scope of this document

4.1.2. Logical Framework

The following figure provides a logical view of components and the interaction between parties in the exchange of healthcare information. Elements illustrated in the figure are explained in the subsections below.

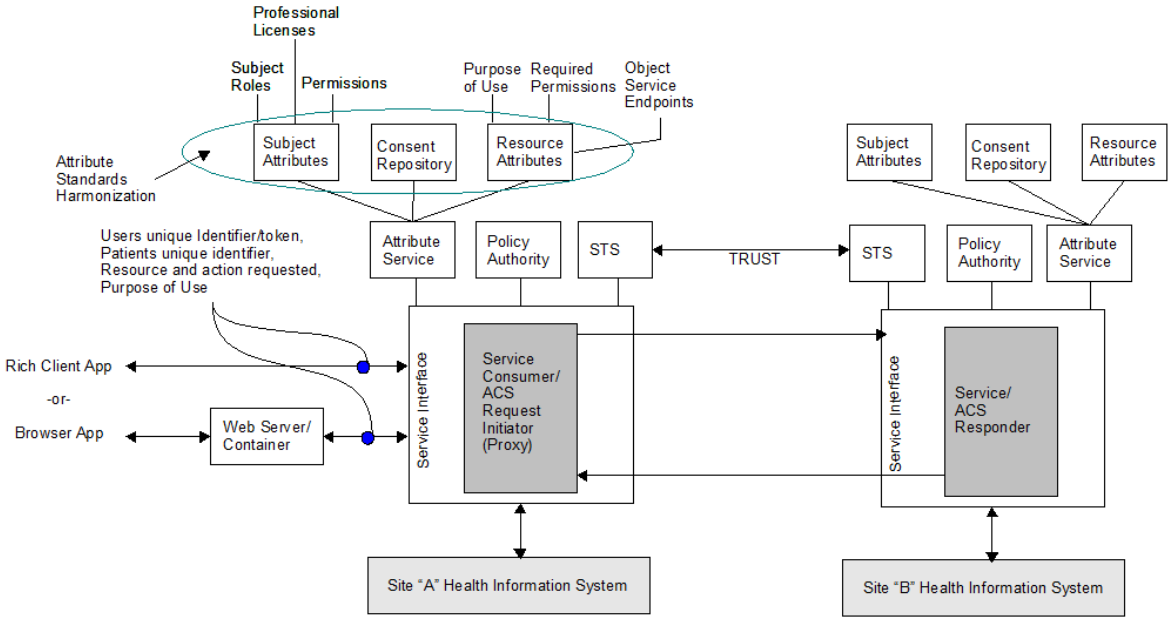

Figure 2 – Logical Framework/Interaction

4.1.2.1. Client Application

Both rich (desktop) and browser based applications are supported in this profile. The client application may ONLY request resources from the local service interface. Client requests must include

- Purpose of Use (POU),
- Object requested as described in **HL7 Version 3 Standard: Role-based Access Control Healthcare Permission Catalog [HL7-RBAC]**,
- Action requested may be in the form of create, read, update, delete, and edit.
- Additional actions may be passed if agreed upon by both parties.
- Resource (patient unique identifier if known, and the user's unique identifier or token.

4.1.2.2. Service Interface

This interface acts as a bus to all available services within a given entity. Users may only request allowed resources through this interface. The responding service interface at the remote entity will be responsible for authenticating the validity of the token by parsing out any assertions (SAML Assertion) made and providing them to the ACS so an access control decision may be made.

The responding service interface will be responsible for providing any object translations required within that entity. An example for consideration is shown in the following table.

HL7 Object	Site A – Service	Result Format	Site B – Service	Native Format
Radiology Order (common)	Http://siteA/XSPA/Radiology?wsdl	CDA	Http://siteB/XSPA/Radiology?wsdl	CDA
Radiology Order (site specific)	Http://siteA/ClinServices/Radiology?wsdl	CDA	Http://siteB/Orders/Radiology?wsdl	Proprietary

In the above table, Site B would be responsible for translating from the proprietary format of a radiology order to Clinical Documentation Architecture (CDA).

Note: This profile DOES NOT provide for or recommend any clinical data exchange standards.

4.1.2.3. Access Control System

In typical implementations, no user or application is allowed direct access to patient information. The XSPA profile of WS-Trust supports this requirement by sending all requests through an Access Control System (ACS). In performing the request, the ACS may acquire additional attribute information related to a user's location, purpose of use, permissions, roles, license information, remote service endpoints, and requested resource requirements and actions. A local access control

decision may be made to deny access to the requested resource (course-grained access control). If permitted the requesting ACS will request a token be granted from an STS that asserts all known information relevant to the request. The requesting ACS is responsible for enforcing the access control decision.

It should be noted that the requesting ACS may make an access control decision to deny access to remote resources based on local internal policies. If the requesting ACS determines to permit a transaction, an applicable token will be requested from an STS and the subsequent request to the remote web service will be made with that token.

4.1.2.3.1. Abstracting Access Control Decisions in Cross Enterprise Implementation

It is assumed that if a set of attributes are evaluated against identical policies that the resultant decision would be the same regardless of the location of the evaluation. This profile in no way defines how or what tools sets will be used in any vendor implementation. Instead it is recommended that abstract program logic be used to describe how access decisions are reached between parties and internally.

The following describes in an abstract manner how required permissions and permission granted to a subject could be evaluated.

```
If permissions.attributeValue equals requiredpermission.attributeValue
then
        permit
else
        deny
end if;
```

4.1.2.4. Attribute Services

The Attribute Service provides the ACS additional attributes necessary to make an access control decision. This profile focuses a healthcare use case and its required attributes to describe its implementation. This by no means limits its implementation to this sector. Any vocabulary could be offered to ACS through the Attribute Service model. Attributes within this profile will be harmonized with and support ISO 10148-3.

4.1.2.4.1. Subject Attributes

This service provides attributes that are specific to the requesting user and are only known to the entity which manages them.

SubjectName

This is the name of the user as required by HIPAA Privacy Rule, Section 164.528 – Accounting of Disclosures SubjectName will be typed as a string in plain text with an identifying element of <urn:oasis:names:tc:SAML:2.0:profiles:attribute:XPSA:subjectname>

SubjectID (Optional)

In some implementations, a country may require a government issued identifier. For example, in the U.S., the subjects National Provider Identifier (NPI) is required for all HIPPA standard transactions. SubjectID will be typed as a string in plain text with an identifying element of <urn:oasis:names:tc:SAML:2.0:profiles:attribute:XPSA:subjectid>.

Organization

This is the organization that the user belongs to as required by HIPAA Privacy Disclosure Accounting. Organization will be typed as a string in plain text with an identifying element of <urn:oasis:names:tc:SAML:2.0:profiles:attribute:XPSA:organization>

Role (Optional)

Roles are placeholders for permission defined by that organization. The use of Role as criteria to make an access control decision would require agreement of their vocabulary between organizations, which may be mandated by law or by a multi-organizational agreement, e.g., by members of a health information exchange organization. Permissions, assigned to roles within an organization, are defined in the HL7 Version 3 Standard: Role-based Access Control Healthcare Permission Catalog **[HL7-RBAC]**. Licensing information related to subject playing the role may be asserted in the form of a certificate.

Permission (optional)

There is no explicit assertion of permission required by this profile.

The permission in use is determined by the Action on the Target. See Action below. The Permission is the ANSI INCITS RBAC compliant action-object pair representing the authorization required for access by the protected resource. In this profile, the service acting as a proxy for the user implicitly asserts the authorizations of the principal by sending the request as required by the inter-organizational agreement. The sending of the request by the trusted service consumer is evidence to the relying party (service) that the service consume has:

- Correctly authenticated at the assurance level required by mutual agreement,
- Determined that the user is authorized to make the request and receive the information requested,
- Verified that all subject consent directives have been consulted and there is no prohibition regarding the request.

Figure 3 illustrates the general relationship between subject (user) and granted permissions to specific objects as a relationship to their POU. POU is determined during the login process. Roles in this relationship are placeholders for permissions. Permission defines the object-action relationship.

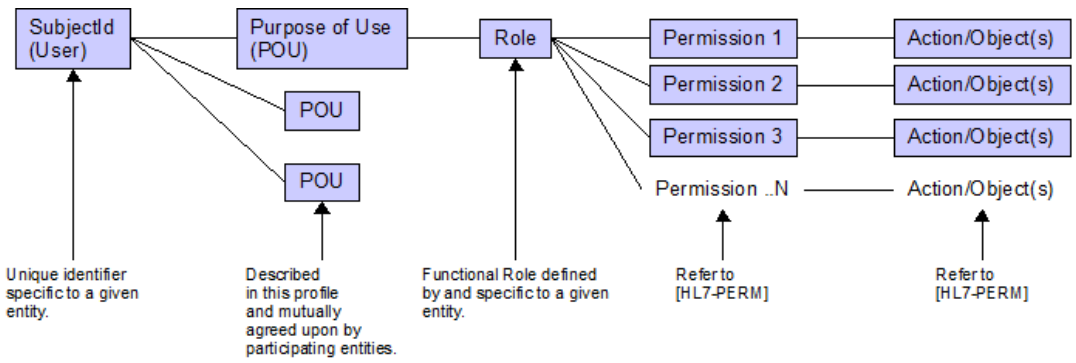

Figure 3 Determining Subject Permissions

4.1.2.4.2. Resource Attributes

Required Permissions

Required permissions for object action pairing are known only to a given entity. A relationship exists with the client application requested resource and the permission required for its action. By determining that the requester's role and asserted POU should be permitted to act on the object requested, an access control decision can be made prior to traversing across enterprises.

Objects, actions and permissions are described in the HL7 Version 3 Standard: Role-based Access Control Healthcare Permission Catalog **[HL7-RBAC]**.

Purpose of Use	Object	Action	Permission Required
Healthcare Access	{R, Radiology Order}	Read	PRD-004
Emergency Access	{R, Radiology Order}	Read	PEA-001
(etc)			

Purpose of Use

Purpose of use (POU) provides essential context for requests to information resources. Each purpose of use will be unique to a specific claim, and will establish the context for other security and privacy attributes. Within a claim, all other claims and assertions must be bound to the purpose of use. POU allows the service to consult its policies to determine if the user's claims meet or exceed those needed to allow access.

From a clinician's perspective, they will interact with the healthcare application to establish the POU at the time of requesting patent information. This will take the form of the healthcare application offering a list of appropriate POUs to the clinician based on their role. (e.g. Healthcare Treatment, Emergency Treatment, etc.).

A patient establishes usage permissions through consent directives, which will be correlated to each POU. For example, a patient's consent directive under normal conditions (Healthcare Treatment POU) will include objects, confidentiality codes and policy attributes which may be much different from those in an emergency (Emergency Treatment POU). Consent directives specific to a purpose of use allows both the patient and the system enforcing access control to share a common context and vocabulary for requirements, claims and policy enforcement that provides for interoperability of both security and privacy enforcement.

The following list of healthcare related purposes of use is specified by this profile.

- Healthcare Treatment
- Emergency Treatment
- System Administration
- Operations
- Payment

- Research
- Marketing

Object Service Endpoints

Additionally, object service endpoints which are shared / distributed across trusted entities are also provided. Participating organization must share clinical service endpoints, have a common method naming convention, and data format exchange standard as described in the following table**.**

Site	Object	Endpoint	Operation	Result Format
Site A	{R, Radiology Order}	Http://siteA/XSPA/ClinicalServices?wsdl	GetRadiologyOrder	CDA
Site B	{R, Radiology Order}	Http://siteB/webservices/Clinical?wsdl	GetRadiologyOrder	CDA
Site C	{R, Radiology Order}	Http://siteC/ws/Radiology?wsdl	GetRadiologyOrder	CDA

Consent Repository

The resource (e.g. patient) may choose to constrain access to their records. They do so by establishing a consent directive which subsequently resides in the consent repository. The responsibility of enforcing these constraints falls on both the Service Responder and Consumer.

Figure 4 shows the general relationship between POU and resources consent directive when constraining permissions to specific subjects (users) or their roles. In this profile POU will be those previously specified in this document. Consent Code in this profile will implement the HL7 Version 3 Consent related vocabulary including Confidentiality Codes [HL7-Consent]. The Subject and Role vocabulary for this profile conforms to the ASTM standard for structural healthcare roles **[ASTM E1986-98(2005)].**

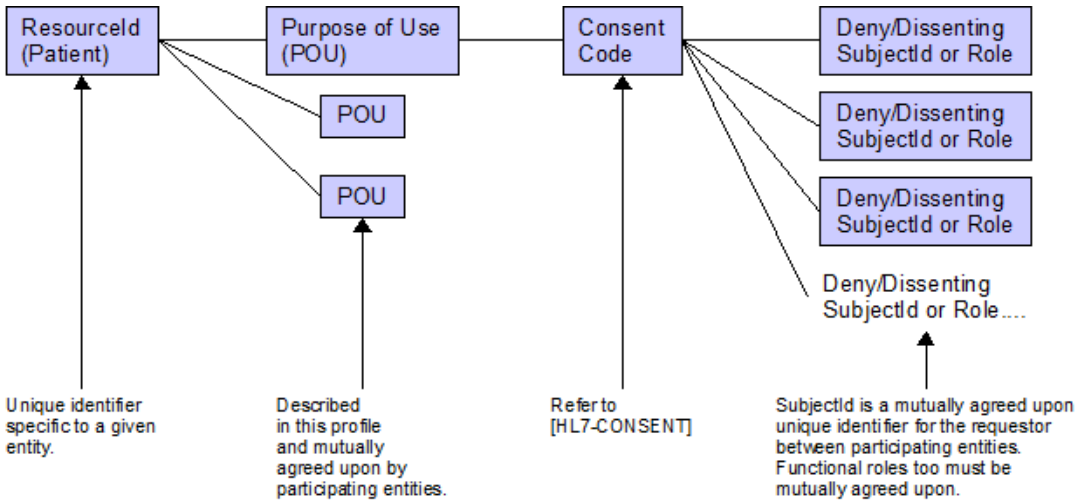

Figure 4 POU and User Based Access (UBA) Control

Figure 5 shows the general relationship between POU and masking of specific object based on the subject or their role. In addition to those vocabularies specified for use with the components illustrated in figure 4, the objects of this profile are defined through the HL7 RBAC Permission Catalog **[HL7-RBAC]**.

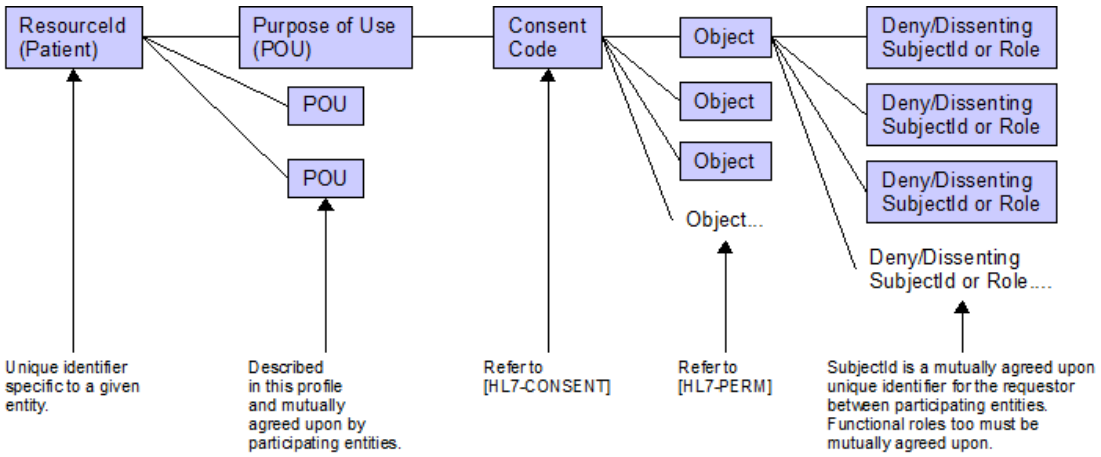

Figure 5 Resource Object Masking (MA)

4.1.2.5. Policy Authority

The Policy Authority contains rules specific to that entity as well as any mutually agreed upon cross-enterprise policies. The ACS sources these policies to make access control determination prior to making any external requests. This document does not address how the policies are represented. It remain up to the implementer what best fits their ACS implementation.

4.1.2.6. STS

The Security Token Service (STS) acts as a brokering agent by issuing tokens, in the form of a response to the requesting ACS. The requesting ACS then submits the token to the web service. The web service may subsequently contact the STS to ensure the validity of the token and its claims (e.g. none of the claims have been revoked).

The most fundamental concept of WS-Trust is that requestors and consumers of tokens trust the issuer of that token: the STS.

4.1.2.7. Patient Lookup Service

Refer to section 4.1.4 for an example of interactions. This document does not address or suggest how this function may be implemented.

4.1.2.8. Transmission Integrity

The XSPA profile of WS-Trust recommends the use of reliable transmission protocols. Where point-to-point transmission integrity is required, this profile specifies the Healthcare Information Technology Standards Panel (HITSP) Secured Communication Channel (TP17).

4.1.2.9. Transmission Confidentiality

The XSPA profile of WS-Trust recommends the use of secure transmission protocols. Where point-to-point transmission confidentiality is required, this profile specifies the HITSP TP17.

4.1.2.10. Error States

This profile will implement messaging standards described in WS-Trust when communicating error states between requesting and responding entities.

4.1.2.11. Security Considerations

<Risk assessment>

4.1.2.12. Confirmation Identifiers

No known common identifiers at this time.

4.1.2.13. Metadata Definitions

This profile will utilize the WS-Trust <AttributeStatement> to inject a SAML assertion into request.

4.1.2.14. Naming Syntax, Restrictions and Acceptable Values

This profile will support the namespace requirements described in WS-Trust.

4.1.2.15. Namespace Requirements

This profile will support the namespace requirements described in WS-Trust.

4.1.2.16. Attribute Rules of Equality

All asserted attributes child to <AttributeStatement> element will be typed as strings. Two <Attributes> elements refer to the same SAML attribute if and only if their Name XML attribute values are equal in a binary comparison.

4.1.3. Example Messages

The following are intended to provide additional guidance to implementers. These examples leverage the draft profile **[XSPA-SAML]** Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of Security Assertion Markup Language (SAML). Implementers may choose to utilize other means of asserting attributes beyond SAML capabilities within WS-Trust.

4.1.4. Example Header

4.1.5. Timestamp Example

Timestamp information should be included – TBD.

4.1.6. Request / Response – Cross Enterprise Patient Lookup

Event Flow

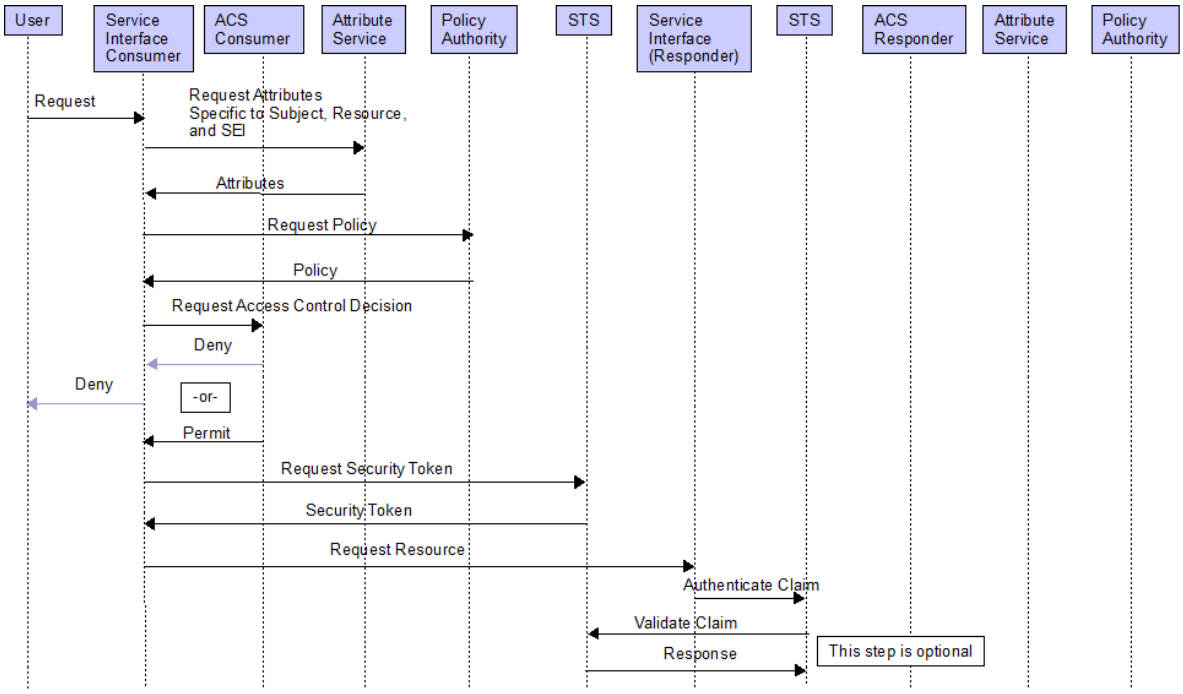

Figure 6 Cross Enterprise Patient Lookup Event Flow

Patient search across enterprises may only require a coarse grain approach to authorization where an access control decision can be made without the evaluation of subject attributes. In this case the responder's services interface may execute the lookup without having to interact with the ACS. This a result of trust between two STSs.

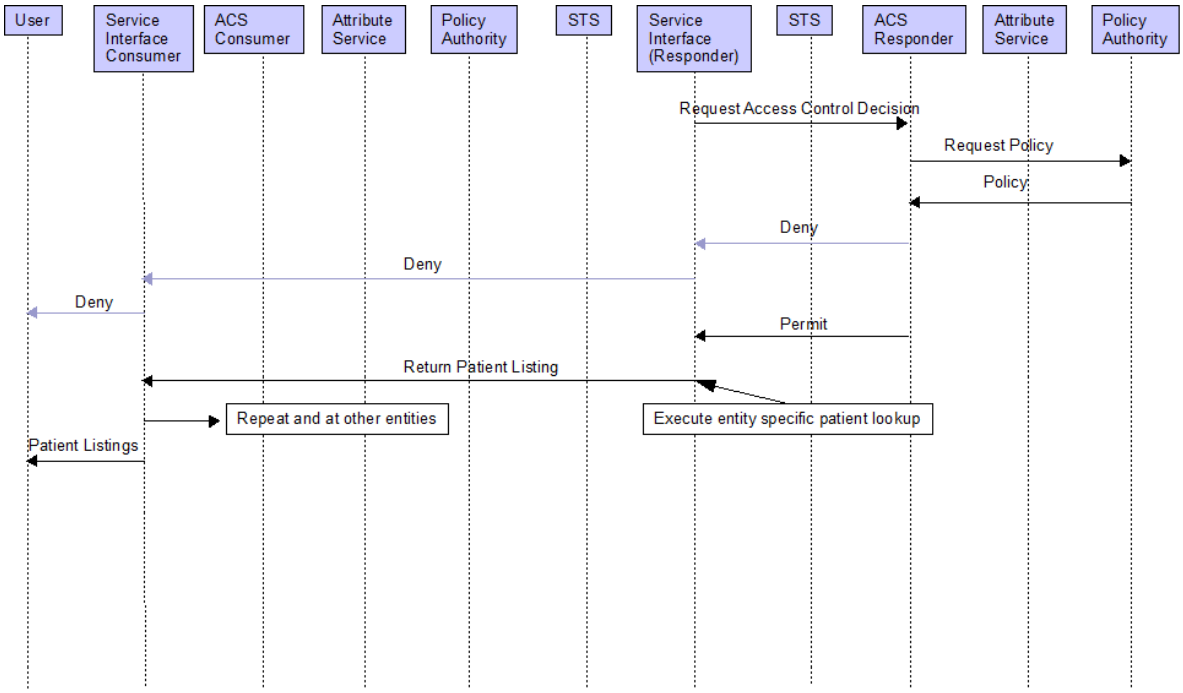

Figure 7 Cross Enterprise Patient Lookup Event Flow (Cont.)

Request

Within this XSPA profile SAML 2.0 assertions are used to establish the attributes of the actors involved in a transaction.

The following is an example of the expected syntax of an RST with respect to this XSPA profile.

```
<wst:RequestSecurityToken Context="..." xmlns:wst="...">
        <wst:TokenType>...</wst:TokenType>
        <wst:RequestType>...</wst:RequestType>
        <wst:SecondaryParameters>...</wst:SecondaryParameters>
        ...
    </wst:RequestSecurityToken>
```

Response Example

This document does not define what attributes of the patient demographic record should be contained in the resultant patient list. At minimum the patientId, name, age, gender, primary clinic, and/or provider should be available for the consumer to make a selection. This also assumes that all requests for a patients clinical record require the selection of that patient first.

The following is an example of the expected syntax of an RSTR with respect to this XSPA profile:

```
<wst:RequestSecurityTokenResponse>
```

```
 <wst:TokenType>
  SAML
 </wst:TokenType>
  <wsp:AppliesTo>
   <wsa:EndpointReference>
    <wsa:Address>http://hostname/webservice/patientlookup</wsa:Address>
   </wsa:EndpointReference>
  </wsp:AppliesTo>
<RequestedSecurityToken>
 <saml:Assertion … >
  <AttributeStatement>
   <Attribute
   AttributeName="subject"
   AttributeNamespace="http://schemas.xmlsoap.org/claims">
    <AttributeValue>
      <Attribute AttributeName=”subjectName”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Doctor, Bob
     </AttributeValue>
      </Attribute>

     <!-- optional -->
      <Attribute AttributeName=”subjectID”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       100011
     </AttributeValue>
      </Attribute>
     <!-- end optional -->

      <Attribute AttributeName=”organization”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Fort Harrison, Helena MT
     </AttributeValue>
      </Attribute>

     <!-- optional -->
      <Attribute AttributeName=”Role”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Physician
     </AttributeValue>
      </Attribute>
      <Attribute AttributeName=”permissions”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       PRD-001
     </AttributeValue>
     <AttributeValue>
       PRD-004
     </AttributeValue>
     <AttributeValue>
       PRD-011
     </AttributeValue>
      </Attribute>
     <!-- end optional -->
```

```
   <AttributeStatement>
  <Attribute
   AttributeName="resource"
   AttributeNamespace="http://schemas.xmlsoap.org/claims">
   <AttributeValue>
      <Attribute AttributeName=”resourceName”
    <AttributeValue>
      patientSearch
      </AttributeValue>
    </Attribute>
   <Attribute AttributeName=patientName
   <AttributeValue>
      Doe,John
      </AttributeValue>
    </Attribute>
   <Attribute AttributeName=”action”
   AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
      <AttributeValue>
         Read
      </AttributeValue>
       </Attribute>

   </AttributeStatement>
  </saml:Assertion>
 </wst:RequestedSecurityToken>
</wst:RequestedSecurityTokenResponse>
```

Required Attributes

Attribute	Namespace	Comments
Patientidentifier (lastname,firstname,middle initial, etc.)	tbd	Each entity may employ varying methodologies for patient lookup. Participating parties will need to agree on what is required and to what level of detail.
Subjectname	tbd	
Organization	tbd	
Purposeofuse	tbd	
Resourcename	tbd	Refer to [HL7-PERM] for object values
Action	tbd	Create, read, update, delete, edit, and other mutually agreed upon action values.

Optional Attributes

Attribute	Namespace	Comments
SubjectId	tbd	For U.S. Implementations this is the users Medicaid ID.
Role	tbd	
Permission	tbd	

4.1.7. Request / Response – Medical Record Access

Event Flow

Refer to Figure 6 for local access control decision. Figure 8 shows the access control decision event at remote site. In this example the ACS must be provided with the patient's consent directives.

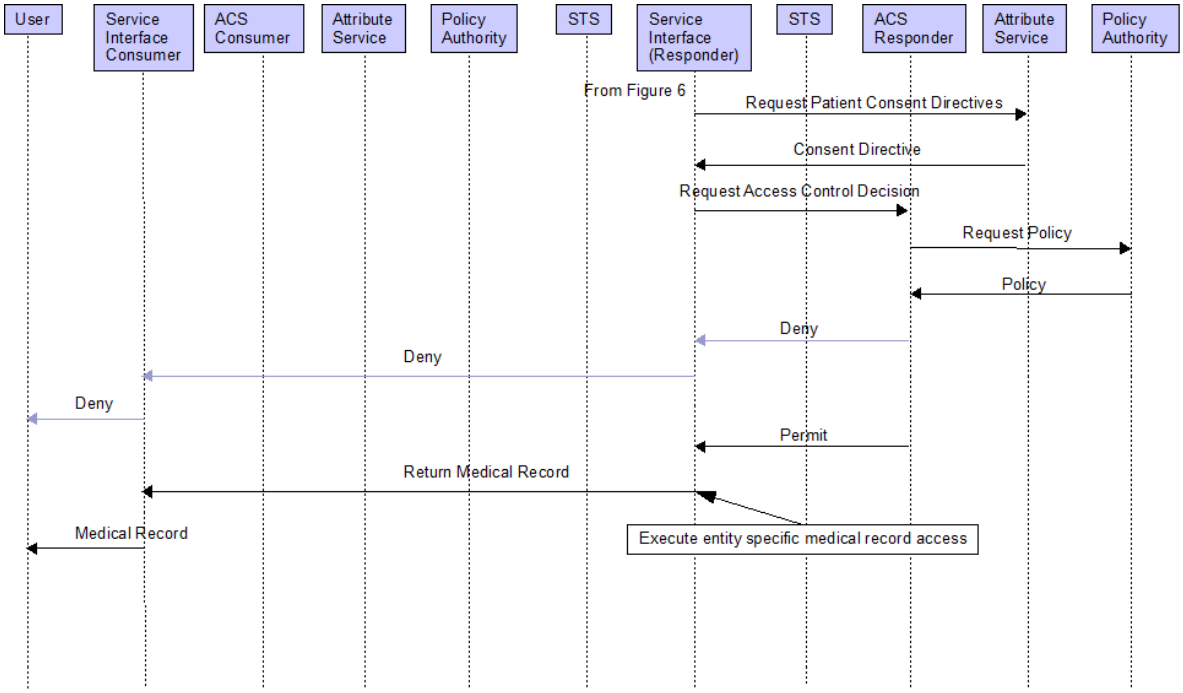

Figure 8 Medical Record Event Flow

Request Example

The following is an example of the expected syntax of an RST with respect to this XSPA profile with requesting access to patient medical record(char).

```
<wst:RequestSecurityToken Context="..." xmlns:wst="...">
        <wst:TokenType>...</wst:TokenType>
        <wst:RequestType>...</wst:RequestType>
        <wst:SecondaryParameters>...</wst:SecondaryParameters>
        ...
    </wst:RequestSecurityToken>
```

Response Example

The following is an example of the expected syntax of an RSTR with respect to this XSPA profile with requesting access to patient medical record(char).

```
<wst:RequestSecurityTokenResponse>
 <wst:TokenType>
   SAML
 </wst:TokenType>
  <wsp:AppliesTo>
   <wsa:EndpointReference>

<wsa:Address>http://hostname/webservice/MedicalRecord/Radiology</wsa:Address>
   </wsa:EndpointReference>
  </wsp:AppliesTo>
<RequestedSecurityToken>
 <saml:Assertion … >
  <AttributeStatement>
   <Attribute
   AttributeName="subject"
   AttributeNamespace="http://schemas.xmlsoap.org/claims">
    <AttributeValue>
      <Attribute AttributeName=”subjectName”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Doctor, Bob
     </AttributeValue>
      </Attribute>

     <!-- optional -->
      <Attribute AttributeName=”subjectID”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       100011
     </AttributeValue>
      </Attribute>
     <!-- end optional -->

      <Attribute AttributeName=”organization”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Fort Harrison, Helena MT
     </AttributeValue>
      </Attribute>

      <Attribute AttributeName=”role”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       Physician
     </AttributeValue>
      </Attribute>
      <Attribute AttributeName=”permissions”
  AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
     <AttributeValue>
       PRD-001
     </AttributeValue>
     <AttributeValue>
       PRD-004
     </AttributeValue>
     <AttributeValue>
       PRD-011
     </AttributeValue>
      </Attribute>
```

```
  <AttributeStatement>
  <Attribute
   AttributeName="resource"
   AttributeNamespace="http://schemas.xmlsoap.org/claims">
   <AttributeValue>
      <Attribute AttributeName=”resourceName”
    <AttributeValue>
     RadiologyReport
     </AttributeValue>
    </Attribute>
   <Attribute AttributeName=patientId
   <AttributeValue>
      Doe,John
     </AttributeValue>
    </Attribute>
   <Attribute AttributeName=”action”
   AttributeNameSpace=”http://schemas.xmlsoap.org/claims”>
      <AttributeValue>
         Read
      </AttributeValue>
       </Attribute>

   </AttributeStatement>
  </saml:Assertion>
 </wst:RequestedSecurityToken>
</wst:RequestedSecurityTokenResponse>
```

Required Attributes

Attribute	Namespace	Comments
PatientId	tbd	Unique to remote entity
Subjectname	tbd	
Organization	tbd	
Purposeofuse	tbd	
Resourcename	tbd	Refer to [HL7-PERM] for object values
Action	tbd	Create, read, update, delete, edit, and other mutually agreed upon action values.
Permission	tbd	
Role	tbd	

Optional Attributes

Attribute	Namespace	Comments

PatientName	tbd	

4.1.8. Masking of Clinical Data

In masking clinical data, the event flow is identical to accessing a patient-specific medical record discussed in section 4.1.5 with the exception that the patient's consent directive requires masking of portions of the clinical record**.** The response in this case will need to contain an obligation defining which object must be hidden from the requesting user. The consuming ACS and its service interface must enforce this obligation.

4.1.9. Enforcement Cross Enterprise Business Rules

In enforcing cross enterprise business, the event flow is identical to accessing a patient specific medical record as discussed in section 4.1.5 with the exception that entities internal policies requires additional controls on the clinical record**.** The response in this case will need to contain a policy snippet defining the business rule. The consuming ACS and its service interface must enforce this business rule.

4.1.10. Request for Additional Attributes

Event Flow

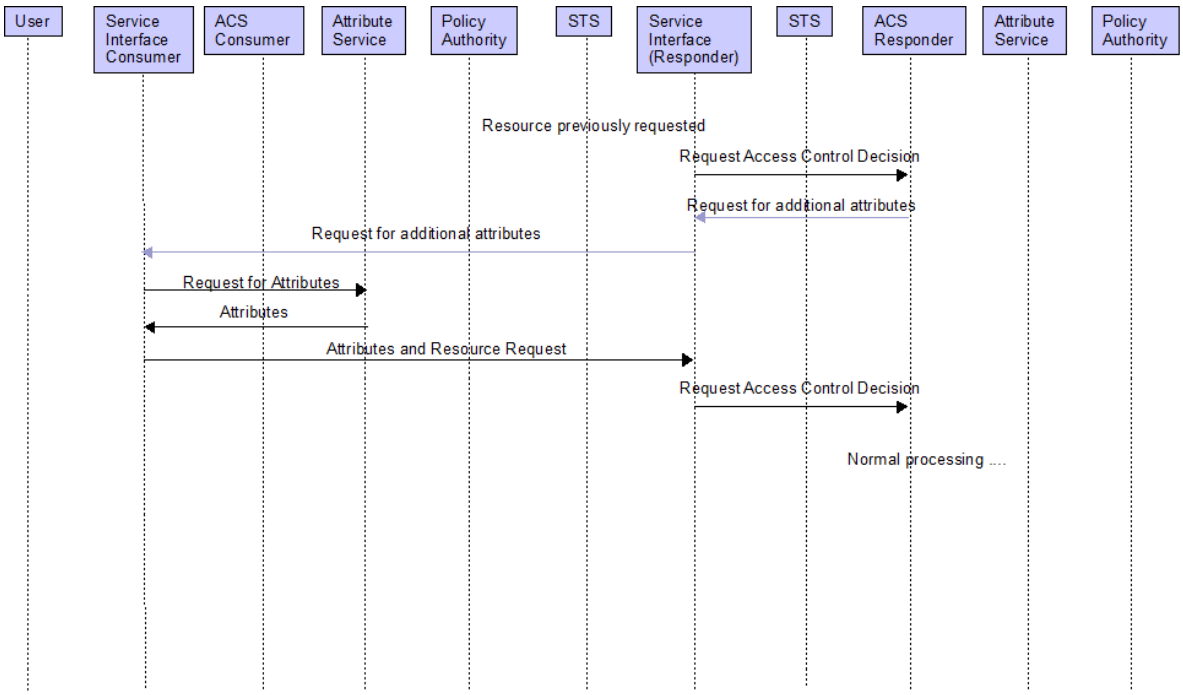

Figure 9 Request of Additional Attributes

5. References

[ASTM E1986-98(2005)] Standard Guide for Information Access Privileges to Health Information.

[HITSP/TP20] Healthcare Information Technology Standards Panel Security and Privacy Technical Committee. *HITSP Access Control Transaction Package, v1.1*. October 2007.

[HL7-Consent] Healthcare Level 7. RBAC Consent related vocabulary including Confidentiality Codes, v3.0.

[HL7-RBAC] Healthcare Level 7. *RBAC Healthcare Permission Catalog, v3.38*. November 2007.

[SAMLPROF] Organization for the Advancement of Structured Information Standards. *Profiles for the OASIS Security Assertion Markup Language, v2.0*, February 2005.

[WS-Trust] Organization for the Advancement of Structured Information Standards. *WS-Trust, v1.3*, March 2007.

[XSPA-SAML] Draft Profile - Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of Security Assertion Markup Language (SAML)