GFD-R-P.157
OGSA-Authz

David Chadwick, University of Kent
Linying Su, University of Kent
13 November 2009

Use of WS-TRUST and SAML to access a Credential Validation Service

Status of This Document

This document provides information to the Grid community about a proposed standards track protocol. Distribution is unlimited.

Copyright Notice

Abstract

This document provides a protocol for an authorization component to access an external credential validation service (CVS) prior to calling a policy decision point (PDP). The protocol is a profile of a SAML [SAML] attribute assertion carried by WS-Trust [WSTRUST].

Contents

1. Introduction

This document describes a protocol for accessing a credential validation service (CVS) by the context handler of a policy enforcement point (PEP) or a policy decision point (PDP). It is based on the model in [ARCH]. The CVS is a necessary functional component in authorization which performs the task of validating the user's presented credentials before the valid attributes (extracted from the credentials) are used by the policy decision point (PDP) in order to make an access control decision. The outer protocol is WS-Trust [WSTRUST]. This tells the CVS how to operate (e.g. in push or pull mode) and to return XACML formatted attributes. The payload of WS-Trust is a single SAML assertion created by the context handler of the PEP or PDP, This SAML assertion contain zero or more attribute statements, each attribute statement being an authorisation credential that the PEP has received and which needs validating. This protocol allows tokens/credentials to be presented in any format to the CVS, but always returns tokens formatted as XACML attributes, so that they are ready for submission to a PDP that talks the XACML request context [OGFXACML].

2. Notational Conventions

The key words 'MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" are to be interpreted as described in RFC 2119 [BRADNER1]

3. Model and Definitions

The authorization architecture is described in [ARCH]. Figures 1 to 4 are simplified versions of the figures in [ARCH] and show the different ways in which the CVS access protocol might be used. The protocol might be called by the context handler in either the PEP or the PDP, and might carry the authenticated name of the subject with or without a bag of credentials, and with or without references to various credential issuing services (CISs) that should be contacted to pull credentials.

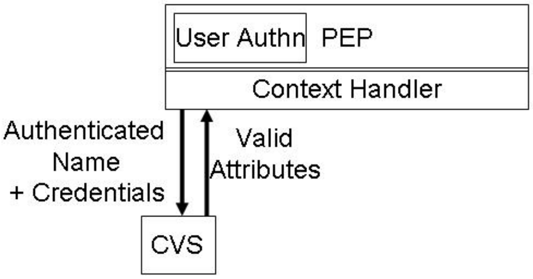

Fig 1 PEP Context Handler – Push Credentials

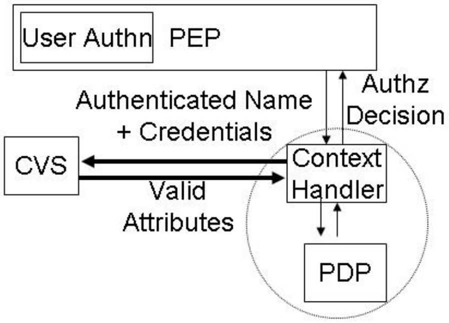

Fig 2 PDP Context Handler – Push Credentials

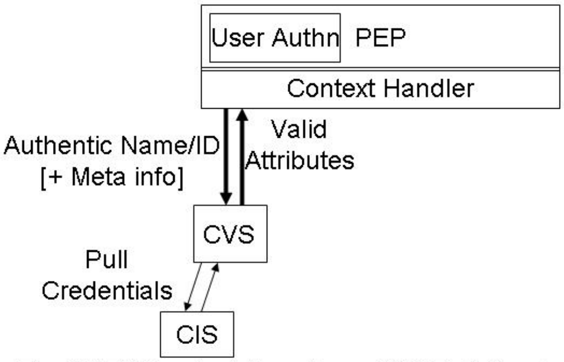

Fig 3 PEP Context Handler – CVS Pull Credentials

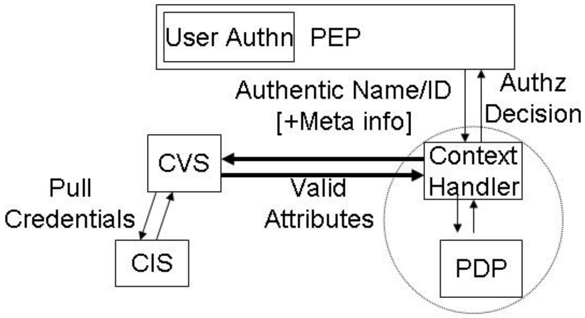

Fig 4 PDP Context Handler – CVS Pull Credentials

The PEP may provide any arbitrary set of credentials (e.g. 'member of university X', 'member of grid project Y', 'registered doctor', 'certified engineer' etc.) issued by any arbitrary set of attribute authorities (AAs) or credential issuers, in any standard format (e.g. SAML assertion, X.509 AC etc). The PEP may also provide an arbitrary set of references (meta-information) to credential issuing services. This document does not specify how the PEP obtains these credentials or CIS meta-information references; they might be provided by the end user, or the end user's client software, or by another component of the authorization infrastructure, such as an out of band meta-data transfer service.

The target resource will only trust a limited set of CISs[1], and these trust relationships will be configured into its Credential Validation Service (CVS) in the form of a Credential Validation Policy. A CVS will validate the presented credentials according to its configured Credential Validation Policy, and will return the set of valid attributes (in XACML format) to the PEP. The PEP may then pass these to a PDP that supports the XACML request context [OGFXACML] for it to make an access control decision.

Note that it is not within the scope of this document to define the contents of the credential validation policy, but it might contains such rules as "University X is trusted to issue doctoral degrees to anyone", "Steve Jones is trusted to say who is a member of Project Y". "Gold credentials > Silver credentials > Bronze credentials". "No credentials can be older than n minutes". "Steve Jones can delegate issuing member credentials to any project manager within Project Y" etc.

3.1 Credential Push vs. Credential Pull

The CVS can operate in three ways – credential push mode, credential pull mode or both modes.

- In credential push mode, the requestor will pass the credentials that it has received from the user to the CVS for validation. For example, this is the way that the original VOMS system was designed to work, although VOMS [VOMS] does not specifically refer to "a CVS" as the module that validates the pushed VOMS ACs.
- In credential pull mode, the CVS is passed the DN of the authenticated user and it pulls the credentials of this user from the various CISs, and validates them. For example, this is the way that a previous version of GridShib (version 0.5.2) worked,[2] when the Shibboleth Policy Information Point (PIP) goes to fetch the Shibboleth attributes of the user from the Shibboleth Identity Provider (IdP).
- A CVS may operate in both modes simultaneously, being given one set of user credentials, and pulling more in order to fully process the first set. For example, the PERMIS [PERMIS] CVS can operate like this.

The only difference from the protocol initiator's perspective, is that in push mode the request message contains a bag of credentials, in pull mode it does not contain any credentials and may contain references to one or more CISs, and in both modes the request message will contain a bag of credentials and may contain a set of CIS references. An advisory field in the WS-Trust request indicates which mode should be used by the CVS.

The output of all three modes of operation is always the same – the CVS returns a set of valid attributes to the PEP, that are in the defined XACML format for passing to the PDP. Note that 'valid' in this context means attributes that are trusted according to the credential validation policy configured into the CVS. If this policy is changed, then a different set of attributes may well be returned for an identical request.

[1] For example, some shop keepers trust Visa, some trust Amex, and some trust both. The credentials are always authentic in each shop, but they are not always valid.

[2] Like VOMS, the later versions of GridShib (version 0.6.0 onwards) are focused on attribute push.

3.2 Relationship of CVS to Security Token Service (STS) and PIP

WS-Trust [WSTRUST] version 1.3 is a specification by OASIS, supported by Microsoft, IBM and others[3] that enables security token interoperability by defining a request/response SOAP protocol whereby clients can request from some trusted authority that a particular security token be exchanged for another one. The security token service (STS) is the trusted authority that responds to WS-Trust requests.

Madsen[4] identifies that an STS actually has three different functionalities, namely: security token exchange, security token issuing and security token validation. The last two functions are special simplified cases of the first. In this document we are only interested in the third piece of functionality, security token validation. Therefore we have decided to give this specialized functionality its own name – the credential validation service (CVS) – rather than the generic name STS, since STS implies a much greater functionality than that which is required here. In general then, an STS can accept security tokens in multiple formats and output security tokens in multiple formats. What the grid authorization infrastructure requires is that the CVS can be given security tokens (or credentials) in multiple formats but always returns them as valid attributes in XACML format.

XACML [XACML] is a specification by OASIS that defines a language for expressing access control policies in XML. XACML does not define or even mention security tokens or credentials. The nearest it comes is to define a Policy Information Point (PIP) as the system entity that acts as a source of (asserted) attribute values. Since the CVS described in this document is a source of attribute values that are ready to be passed to an XACML conformant PDP, then one can consider that the CVS is a specialized type of PIP that can process credentials and/or security tokens according to a credential validation policy, and that can return valid attributes in exchange for the input credentials.

4. CVS Request Protocol

The request message comprises a single SAML attribute assertion embedded in a WS-Trust request protocol message. Both the WS-Trust request and the SAML attribute assertion give directives to the CVS to tell it how to validate the user's credentials. The attribute values of the SAML attribute statements are used to push the user's credentials to the CVS. The Advice element of the SAML assertion is used to tell the CVS which external attribute authorities (AAs) to contact in order to pull the user's credentials.

The SAML assertion also carries the distinguished name of the user, the distinguished name of the authorization component making the request, and optionally the maximum validity period to attach to the returned XACML attributes.

4.1 SAML attribute assertion profile

This profile is based on SAMLv2 [SAML].

The SAML assertion MUST contain the following fields:

(1) The MajorVersion attribute MUST be set to "2.0"

[3] The WS-Trust specification is available from ftp://www6.software.ibm.com/software/developer/library/ws-trust.pdf

[4] Paul Madsen "WS-Trust: Interoperable Security for Web Services" Available from http://www.xml.com/pub/a/ws/2003/06/24/ws-trust.html

(2) The Issuer element is mandatory and SHOULD contain the name of the authorization component (i.e. the context handler of the PEP or PDP) that is sending this assertion to the CVS. The Format MUST be X.509SubjectName. Where SSL/TLS is used for mutual authentication, then the distinguished name (DN) should be the DN from the X.509 certificate used by the SSL/TLS service (see Section 7). The use of any other format is not specified in this profile.
(3) Ds-signature is optional (see Section 7).
(4) The Subject element MUST be present. The NameID element MUST be used, and the Format MUST be X.509SubjectName. This is the name of the subject whose credentials are to be validated. It will typically be the distinguished name of the subject extracted from the user's end entity certificate (found at the head of the proxy certificate chain that was sent to the PEP). **The CVS should only return the valid attributes of this subject.** The use of any other format is not specified in this profile. The SubjectConfirmation element SHOULD NOT be present, but if it is present this profile does not specify how it is to be used.
(5) The Conditions element is optional. This profile does not specify the use of this element.
(6) The Advice element MAY be present, and if present SHOULD contain an IDPList element (see clause 6). The CVS is free to ignore or use this advice as it wishes. For example, the CVS may contact just the IDPs pointed to in this Advice, or may contact less than this set, or more than this set.
(7) A set of zero or more AttributeStatements MAY be present.

Each AttributeStatement MUST contain at least one Attribute element and SHOULD NOT contain any EncryptedAttribute elements (see Section 7).

The Name attribute of the Attribute element SHOULD indicate the type of credential that is being passed for validation, and the AttributeValue element SHOULD contain the credential. These credentials may be in a variety of formats e.g. X.509 public key certificate, X.509 attribute certificate, X.509 proxy certificate, VOMS X.509 attribute certificate (extracted from or embedded in a proxy certificate), Kerberos Ticket, Shibboleth attribute, proprietary credentials etc.

This profile defines a set of encodings for a variety of binary and other credentials, so that they can be passed to the CVS and recognized by the CVS before decoding and validation commences. This profile does not restrict the type of credentials that can be validated by a CVS, and users are free to define additional types of credentials as Attribute Names.

Credential	Attribute Name	AttributeValue	Comment

X.509 public key certificate of subject (which may be a proxy certificate)	http://www.ietf.org/rfc/rfc4523.txt#userCertificate	Base 64 encoding of the certificate	The CVS will need to parse the PKC and extract the extensions to see what credentials are embedded in it e.g. as subjectDirectoryAttributes or VOMSacseq (1.3.6.1.4.1.8005.100.100.5)
X.509 attribute certificate	urn:oid:2.5.4.58	Base 64 encoding of the attribute certificate	May be a VOMS X.509 AC extracted from a proxy certificate or may be an X.509 AC generated by another Attribute Authority
X.509 public key certificate of a CA	http://www.ietf.org/rfc/rfc4523.txt#cACertificate	Base 64 encoding of the certificate	This PKC does not carry a user credential but may be needed by the CVS to validate the signatures on the received credentials
SAMLv1.0 Assertion	urn:oasis:names:tc:SAML:1.0:assertion	The SAML assertion in XML	The SAMLv.1.0 credential is sent as the attribute value of the encapsulating SAMLv.2.0 assertion
SAMLv1.1 Assertion	urn:oasis:names:tc:SAML:1.0:assertion	The SAML assertion in XML	The SAMLv.1.1 credential is sent as the attribute value of the encapsulating SAMLv.2.0 assertion
SAMLv2.0 Assertion	urn:oasis:names:tc:SAML:2.0:assertion	The SAML assertion in XML	The SAMLv.2.0 credential is sent as the attribute value of the encapsulating SAMLv.2.0 assertion

Table 1. Encodings of various types of credentials as attributes

The identification of Kerberos tokens is specified in [Kerb].

4.2 WS-TRUST request profile

This profile is based on the WS-Trust specification version 1.3 [WSTRUST].

The <wst:RequestSecurityToken> element is used to request the validation of a bag of credentials. This element MUST contain the following fields:

(1) The Context attribute is optional. If present it MUST contain a URI identifying this request. The corresponding response will then carry the same Context attribute so that the requestor can correlate the request and response.

(2) wst:TokenType describes the type of security token being requested and is specified as a URI. The WS-Trust security token type MUST be set to the SAML XACML profile, defined in [SAMLPROF] as

urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML

(3) wst:RequestType is used to indicate the class of function that is being requested and is specified as a URI. The WS-Trust request type for validating credentials by a CVS MUST be set to

http://schemas.xmlsoap.org/ws/2005/02/trust/validate

(4) wst:Claims carries the SAML attribute assertion defined in Section 4.1.

(5) The Dialect attribute of the Claims (wst:Claims@Dialect) has a URI value, which SHOULD indicate the level of service (<push>, <pull> or <pullpush>) that is to be carried out by the CVS.

http://www.ogf.org/authz/2008/06/CVS/push
http://www.ogf.org/authz/2008/06/CVS/pull
http://www.ogf.org/authz/2008/06/CVS/pullpush

If the embedded SAML assertion contains zero attribute statements then the value <pull> MUST be used. If the SAML assertion contains one or more attribute statements then the value <push> or <pullpush> MUST be used. If the SAML assertion contains a SubjectAttributeReferenceAdvice then the value <pull> or <pullpush> MUST be used. Note that the Dialect attribute is advisory only, in that the CVS may need to pull further credentials in order to validate the ones that were pushed even if <pullpush> was not specified, or the CVS may be unable to contact the referred to CISs even if <pull> was requested.

5. CVS Response Protocol

The response message comprises a single SAML attribute assertion, holding the set of valid XACML attributes, embedded in a WS-TRUST response message.

5.1 SAML attribute assertion profile

This profile is based on SAMLv2 [SAML].

The single SAML assertion SHOULD contain the following fields:

(1) The MajorVersion attribute MUST be set to “2.0”.
(2) The Issuer element is mandatory and contains the name of the CVS. The Format MUST be X.509SubjectName. Where SSL/TLS is used for mutual authentication, then the distinguished name (DN) should be the DN from the X.509 certificate used by the SSL/TLS service (see Section 7). The use of any other format is not specified in this profile.
(3) Ds-signature is optional (see section 7).
(4) The Subject element MUST be present. The NameID element MUST be used, and the Format MUST be X.509 subject name. The NameID element MUST contain the DN of the user whose valid XACML attributes are being returned and SHOULD strongly match the Subject field of the request message. The use of any other format is not specified in this profile. The SubjectConfirmation element SHOULD NOT be present, but if present this profile does not specify how it is to be used.

(5) The Conditions element is mandatory if one or more AttributeStatements are present, and MUST contain the NotBefore and NotOnOrAfter attributes, which specify the validity time of the returned XACML attributes. This time period MUST be the intersection of the time periods in the validated credentials. This information may then be used by the PEP to limit the duration of the user's session, before asking the CVS to re-validate the user's credentials. The other elements of the Conditions elements SHOULD NOT be present, but if present, this profile does not specify their use..
(6) The Advice element SHOULD NOT be present, but if present MAY be ignored by the PEP.
(7) A set of zero or more AttributeStatements MUST be present.

Each AttributeStatement MUST contain the following fields
(1) It MUST contain at least one Attribute elementand SHOULD NOT contain EncryptedAttribute elements (see Section 7).
(2) Each Attribute element MUST be encoded according to the XACML Attribute Profile specified in section 8.5 of [SAMLPROF].

5.2 WS-TRUST response profile

This profile is based on the WS-Trust specification version 1.3 [WSTRUST].

The <wst:RequestSecurityTokenResponse> element is used to return a security token or a response to a security token request. This element MUST contain the following fields:

(1) The Context attribute MUST be present if it was present on the request, and must contain the same value, otherwise it MUST be absent.

(2) wst:TokenType describes the type of security token being returned and is specified as a URI. The WS-Trust security token type MUST be set to the SAML XACML profile, defined in [SAMLPROF] as

urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML

(3) wst:RequestedSecurityToken MUST be present if the status code is set to valid, or MUST NOT be present if the status code is set to invalid. If present it MUST contain the single SAML attribute assertion described in Section 5.1
(4) wst:Status. The wst:Code MUST be set to either valid or invalid as defined in Section 9 of [WS-TRUST]. Wst:Reason MAY be set.

```
   <wst:RequestSecurityTokenResponse Context="..." xmlns:wst="
http://schemas.xmlsoap.org/ws/2004/04/trust">
        <wst:TokenType>
urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML
         </wst:TokenType>
        <wst:RequestedSecurityToken>...</wst:RequestedSecurityToken>
        <wst:Status>
         <wst:Code>
http://schemas.xmlsoap.org/ws/2005/02/trust/status/valid
         </wst:Code>
        </wst:Status>
    </wst:RequestSecurityTokenResponse>
```

Figure 5. An example WS-Trust response

6. Element <IDPList>

The <IDPList> element, is defined in Section 3.4.1.3 and 3.4.1.4 of the SAML standard [SAML]. It specifies a list of IDPs (i.e. CISs). Its purpose is to advise the CVS which IdPs are most likely to have attributes for the subject when working in the *credential pull mode (or pullpush mode)* of operation. The CVS should consult its own policy and/or IdP meta information to finally determine which IdPs it will contact and which type of attribute assertion they issue.

The following schema fragments define the <IDPList> and <IDPEntry> elements, and have been copied from the SAML standard for the benefit of the reader).

```
<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"/>


<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>
```

Implementations may use the <IDPEntry> and <GetComplete> elements in an application dependent manner to indicate which CISs should be contacted.

This specification only standardizes one value for the <IDPList> element, which is zero occurrences of the <GetComplete> element and one occurrence of the <IDPEntry> element containing the <ProviderID> element only with the following reserved URI. This URI is used as the value of the <ProviderID> element to tell the CVS to pull all the user's attributes from all known CISs as configured into the CVS's meta data and policy.

http://schemas.ogf.org/ogsa-authz/2008/09/IDPList/GetComplete

Note that the <IDPList> construct on its own does not inform the CVS which user attributes may be obtained from which CISs. This latter information will need to be obtained from the policy and meta-information that is configured into the CVS. This information will need to contain details of, for example: how to connect to the CIS, what credentials should be used, which attributes it is trusted to issue etc. The IDPList primarily serves to restrict which of the CISs should be contacted for the current grid user (i.e. subject), assuming that the CVS has trust relationships with many CISs (e.g. every university in the country).

Note. The meta-information required to contact a CIS at a referenced URI MAY be included in the Loc attribute URI, for example, the authentication method and token that is needed to make a successful connection. If the meta-information is absent from the Loc attribute URI then it is assumed that the meta-information is either already configured into the CVS by out of band means or the CVS knows how to retrieve it.

7. Security Considerations

The requestor and CVS MUST perform mutual authentication of each other, unless a trusted channel is already established between them. Mutual authentication SHOULD be undertaken by transport layer security (TLS/SSL). The recipient SHOULD check that the authenticated DN of the sender of the transport layer message is the same as the DN of the issuer in the received SAML message.

Message confidentiality should be assured between the requestor and the PDP, unless a trusted channel is already established between them. Message confidentiality SHOULD be undertaken by transport layer security (TLS/SSL).

8. Contributors

David W. Chadwick
The Computing Laboratory
University of Kent
D.W.Chadwick@kent.ac.uk

Linying Su
The Computing Laboratory
University of Kent
L.Su-97@kent.ac.uk

9. Glossary

AA – Attribute Authority
CIS – credential issuing service
CVS – credential validation service
IdP – Identity Provider
PEP – policy enforcement point
PIP – policy information point
PDP – policy decision point
SSL – secure sockets layer
STS – security token service
TLS – transport layer security

10. Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

11. Disclaimer

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

12. Full Copyright Notice

13. References

[BRADNER1] Bradner, S. Key Words for Use in RFCs to Indicate Requirement Levels, RFC 2119. March 1997.

[BRADNER2] Bradner, S. The Internet Standards Process – Revision 3, RFC 2026. October 1996.

[CATLETT] Catlett, C. GFD-1: Grid Forum Documents and Recommendations: Process and Requirements. Argonne, Illinois: Global Grid Forum. April 2002.

[GRIDSHIB] Tom Barton, Jim Basney, Tim Freeman, Tom Scavo, Frank Siebenlist, Von Welch, Rachana Ananthakrishnan, Bill Baker, Kate Keahey. "Identity Federation and Attribute-based Authorization through the Globus Toolkit, Shibboleth, GridShib, and MyProxy". Presented at NIST PKI Workshop, April 2006. Available from http://middleware.internet2.edu/pki06/proceedings/welch-idfederation.pdf

[Kerb] OASIS "Web Services Security Kerberos Token Profile 1.1". OASIS Standard Specification, 1 February 2006

[OGFXACML] David Chadwick, Linying Su, Romain Laborde. "Use of XACML Request Context to access a PDP", OGF GWD-R-P, 25 June 2009

[PERMIS] D.W.Chadwick, A. Otenko "The PERMIS X.509 Role Based Privilege Management Infrastructure". Future Generation Computer Systems, Vol 19, Issue 2, Feb 2003. pp 277-289

[SAML] OASIS. "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0", OASIS Standard, 15 March 2005

[SAMLPROF] OASIS "Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0", OASIS Standard, 15 March 2005

[WSTRUST] OASIS "WS-Trust 1.3", CD 6 Sept 2006

[XACML] "OASIS eXtensible Access Control Markup Language (XACML)" v2.0, 6 Dec 2004, available from http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml

[VOMS] Alfieri, R., Cecchini, R., Ciaschini, V., Dell'Agnello, L., Frohner, A., Lorentey, K., Spataro, F., "From gridmap-file to VOMS: managing authorization in a Grid environment". Future Generation Computer Systems. Vol. 21, no. 4, pp. 549-558. Apr. 2005