David Chadwick, University of Kent
Linying Su, University of Kent
11 June 2008

Use of WS-TRUST and SAML to access a CVS

Status of This Document

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

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

Abstract 1
1. Introduction 2
2. Notational Conventions 2
3. Model and Definitions 2
3.1 Credential Push vs. Credential Pull 3
3.2 Relationship of CVS to STS and PIP 4
4. CVS Request Protocol 5
4.1 SAML attribute assertion profile 5
4.2 WS-TRUST request profile 6
5. CVS Response Protocol 7
5.1 SAML attribute assertion profile 7
5.2 WS-TRUST response profile 8
6. Element <SubjectAttributeReferenceAdvice> 9
7. Security Considerations 10
8. Contributors 10
9. Glossary 10
10. Intellectual Property Statement 10
11. Disclaimer 10
12. Full Copyright Notice 11
13. References 11

1. Introduction

This document describes a protocol for accessing a credential validation service (CVS) by 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 protocol is a profile of a SAMLv2 [SAML] attribute assertion carried by WS-Trust [WSTRUST]. It allows tokens/credentials in 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 the PDP.

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 CISs that should be contacted to pull credentials.

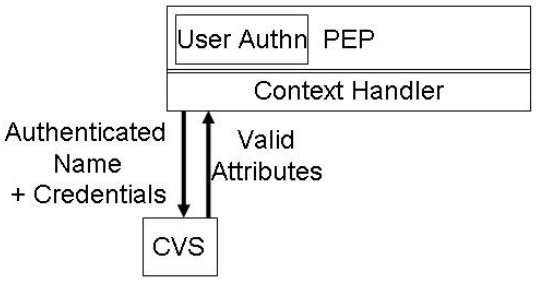

Fig 1 PEP Context Handler – Push Credentials

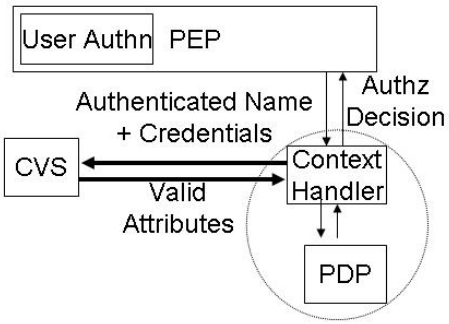

Fig 2 PDP Context Handler – Push Credentials

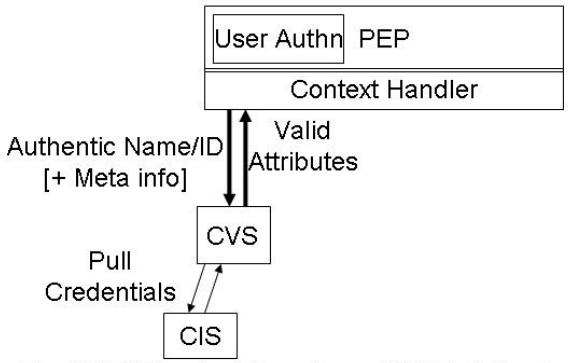

Fig 3 PEP Context Handler – CVS Pull Credentials

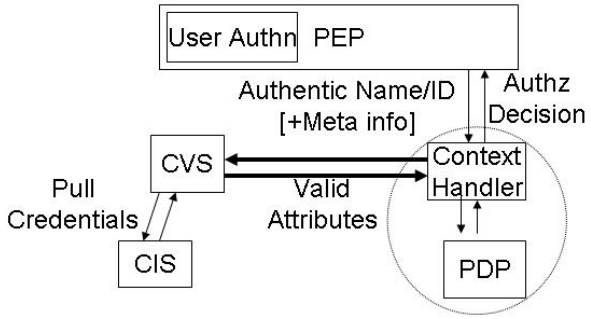

Fig 4 PDP Context Handler – CVS Pull Credentials

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; as well as any 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, but 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 the PDP for it to make an access control decision.

Note. 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

[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.

to the CVS for validation. This is the way that the original VOMS service was designed to work, although 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. This is the way that GridShib works today, when the GridShibPIP goes to fetch the Shibboleth attributes of the user from the Shibboleth IdP. Details of the Shibboleth IdP (the CIS) are currently hardcoded into the GridShibPIP module, rather than being dynamically passed as CIS meta-information. An advanced 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. The PERMIS CVS can operate like this. 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 correct 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.

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.

3.2 Relationship of CVS to STS and PIP

WS-Trust [WSTRUST] is a proposal from Microsoft, IBM and others[2] 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[3] 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 proposal from OASIS that defines a language for expressing access control policies in XML. XACML has nothing to say about 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.

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

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

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 SHOULD contain the following fields:

(1) Version SHOULD be set to "2.0". Conformant implementation MUST support version 2.0 and MAY support other versions.
(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 SHOULD be X.509 subject name. 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) Subject MUST be present. The NameID option MUST be used, and the Format SHOULD be X.509 subject name. This is the name of the subject whose credentials are to be validated. It will typically be the name of the subject extracted from the user's proxy certificate 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 present this profile does not specify how it is to be used.
(5) The Conditions element is optional, but if present MUST contain the NotBefore and NotOnOrAfter attributes, which specify the maximum validity period applicable to the returned XACML attributes. The value of this validity period might typically be taken from the user's proxy certificate. The validity time (NotBefore and NotOnOrAfter Conditions of the SAML assertion) of the returned XACML attributes MUST NOT exceed the validity time of this request. If no validity time is specified in the request, then the CVS is not constrained in the validity time it may return in its response. This profile does not specify the use of any of the other restrictions elements.
(6) The Advice element MAY be present, and if present SHOULD contain a SubjectAttributeReferenceAdvice statement (see clause 6). The CVS is free to ignore or use this advice as it wishes. For example, the CVS may contact just the AAs 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 SHOULD contain Attribute elements and SHOULD NOT contain 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 VOMS acseq (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:1.1: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

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

4.2 WS-TRUST request profile

This profile is based on the WS-Trust specification [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) Version SHOULD be set to “2.0”. Conformant implementation MUST support version 2.0 and MAY support other versions.
(2) the issuer element is mandatory and contains the name of the CVS. The Format SHOULD be X.509 subject name. 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) Subject MUST be present. The NameID option MUST be used, and the Format SHOULD be X.509 subject name. It MUST contains the DN of the user whose valid XACML attributes are being returned and SHOULD be the same as 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 and MUST contain the NotBefore and NotOnOrAfter attributes, which specify the validity time of the returned XACML attributes. 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. This profile does not specify the use of any of the restrictions elements.
(6) The Advice element SHOULD NOT be present, but if present MAY be ignored by the PEP.
(7) A set of zero or more attribute statements MUST be present.

Each Attribute Statement SHOULD contain the following fields
(1) It SHOULD contain Attribute elements and SHOULD NOT contain EncryptedAttribute elements (see Section 7).
(2) Each Attribute element SHOULD 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 [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>
```

6. Element <SubjectAttributeReferenceAdvice>

The <SubjectAttributeReferenceAdvice> element, originally defined in GFD.66 [GFD.66], supplies a statement that the attributes associated with the specified subject may be obtained from the CIS located at the referenced URI. Its purpose is to advise the CVS as to where it may find attributes for the subject when working in the *credential pull mode* of operation.

<SubjectAttributeReferenceAdvice> is of type SubjectAttributeReferenceAdviceType, which extends the Advice element with a set of SubjectAttributeReferences. Each SubjectAttributeReference contains the following:

Attribute [Any number]
: These elements list the attributes that may be obtained from the CIS located at the referenced URI(s). If this component is absent, then it implies that all attributes can be found at the referenced URI(s).

Reference attribute [Required]
: This attribute provides the URI(s) of the CIS from which the above attributes may be obtained.

The following schema fragment defines the <SubjectAttributeReferenceAdvice> element and its SubjectAttributeReferenceAdviceType complex type:

```
<element name="SubjectAttributeReferenceAdvice"
   type="ogsa-saml2:SubjectAttributeReferenceAdviceType"/>
<complexType name="SubjectAttributeReferenceAdviceType">
     <complexContent>
         <extension base="Advice">
             <choice minOccurs=”0” maxOccurs=”unbounded”>
                 <element ref="ogsa-saml2:SubjectAttributeReference" />
             </choice>
         </extension>
    </complexContent>
</complexType>
```

The following schema fragment defines the <SubjectAttributeReference> element and its SubjectAttributeReferenceType complex type:

```
<element name="SubjectAttributeReference"
type="ogsa-saml2:SubjectAttributeReferenceType"/>
<complexType name="SubjectAttributeReferenceType">
             <sequence>
                 <element ref="saml:Attribute" minOccurs=”0” maxOccurs=”unbounded” />
             </sequence>
             <attribute name="Reference” type=”anyURI” use="required" maxOccurs="unbounded"/>
</complexType>
```

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

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

CIS – credential issuing service
CVS – credential validation service
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.

[GFD.66] Von Welch, Rachana Ananthakrishnan, Frank Siebenlist, David Chadwick, Sam Meder, Laura Pearlman. "Use of SAML for OGSI Authorization", Open Grid Forum, Grid Final Documents, GFD.66. March 2006,

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

[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