            Name Attributes for the GSS-API EAP mechanism
                   draft-ietf-abfab-gss-eap-naming-02

Abstract

   The naming extensions to the Generic Security Services Application
   Programming interface provide a mechanism for applications to
   discover authorization and personalization information associated
   with GSS-API names. The Extensible Authentication Protocol GSS-API
   mechanism allows an Authentication/Authorization/Accounting peer to
   provide authorization attributes along side an authentication
   response. It also provides mechanisms to process Security Assertion
   Markup Language (SAML) messages provided in the AAA response. This
   document describes the necessary information to use the naming
   extensions API to access that information.

Status of this Memo

Copyright Notice

Table of Contents

[1](#). Introduction

The naming extensions [I-D.ietf-kitten-gssapi-naming-exts]to the
Generic Security Services Application Programming interface (GSS-API)
[RFC2743] provide a mechanism for applications to discover
authorization and personalization information associated with GSS-API
names. The Extensible Authentication Protocol GSS-API mechanism
[I-D.ietf-abfab-gss-eap] allows an Authentication/Authorization/
Accounting peer to provide authorization attributes along side an
authentication response. It also provides mechanisms to process
Security Assertion Markup Language (SAML) messages provided in the
AAA response. Other mechanisms such as SAML EC
[I-D.ietf-kitten-sasl-saml-ec] also support SAML assertions and
attributes carried in the GSS-API. This document describes the
necessary information to use the naming extensions API to access SAML
assertions in the federated context and AAA attributes.

2. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Naming Extensions and SAML

   SAML assertions can carry attributes describing properties of the
   subject of the assertion. For example, an assertion might carry an
   attribute describing the organizational affiliation or e-mail address
   of a subject. According to Section 8.2 and 2.7.3.1 of [SAMLCORE],
   the name of an attribute has two parts. The first is a URI
   describing the format of the name. The second part, whose form
   depends on the format URI, is the actual name. GSS-API name
   attributes may take a form starting with a URI describing the form of
   the name; the rest of the name is specified by that URI.

   SAML attributes carried in GSS-API names are named with three parts.
   The first is a URN indicating that the name is a SAML attribute and
   describing the context (Section 4). This URI is followed by a space,
   the URI indicating the format of the SAML name, a space and the SAML
   attribute name. The URI indicating the format of the SAML attribute
   name is not optional and MUST be present.

   SAML attribute names may not be globally unique. Many names that are
   named by URNs or URIs are likely to have semantics independent of the
   issuer. However other name formats, including unspecified name
   formats, make it easy for two issuers to choose the same name for
   attributes with different semantics. Attributes using the federated
   context Section 4 are issued by the same party performing the
   authentication. So, based on who is the subject of the name, the
   semantics of the attribute can be determined.

[4]. **Federated Context**

   GSS-API naming extensions have the concept of an authenticated name
   attribute. The mechanism guarantees that the contents of an
   authenticated name attribute are an authenticated statement from the
   trusted source of the peer credential. The fact that an attribute is
   authenticated does not imply that the trusted source of the peer
   credential is authorized to assert the attribute.

   In the federated context, the trusted source of the peer credential
   is typically some identity provider. In the GSS EAP mechanism,
   information is combined from AAA and SAML sources. The SAML IDP and
   home AAA server are assumed to be in the same trust domain. However,
   this trust domain is not typically the same as the trust domain of
   the service. With other SAML mechanisms using this specification,
   the SAML assertion also comes from the party performing
   authentication. Typically, the IDP is run by another organization in
   the same federation. The IDP is trusted to make some statements,
   particularly related to the context of a federation. For example, an
   academic federation's participants would typically trust an IDP's
   assertions about whether someone was a student or a professor.
   However that same IDP would not typically be trusted to make
   assertions about local entitlements such as group membership. Thus,
   a service MUST make a policy decision about whether the IDP is
   permitted to assert a particular attribute and about whether the
   asserted value is acceptable.

   In contrast, attributes in an enterprise context are often verified
   by a central authentication infrastructure that is trusted to assert
   most or all attributes. For example, in a Kerberos infrastructure,
   the KDC typically indicates group membership information for clients
   to a server using KDC-authenticated authorization data.

   The context of an attribute is an important property of that
   attribute; trust context is an important part of this overall
   context. In order for applications to distinguish the context of
   attributes, attributes with different context need different names.
   This specification defines attribute names for SAML and AAA
   attributes in the federated context.

   These names MUST not be used for attributes issued by a party other
   than one closely associated with the source of credentials unless the
   source of credentials is re-asserting the attributes. For example, a
   source of credentials can consult whatever sources of attributes it
   chooses, but acceptors can assume attributes in the federated context
   are from the source of credentials.

[5](). Name Attributes for GSS-EAP

This section describes how RADIUS attributes received with the GSS-
EAP mechanism are named.

The first portion of the name is urn:ietf:params:gss:radius-attr (a
URN indicating that this is a GSS-EAP RADIUS AVP). This is followed
by a space and a numeric RADIUS name as described by section 2.6 of
[I-D.ietf-radext-radius-extensions]. For example the name of the
User-Name attribute is "urn:ietf:gss:radius-attr 1". The name of
extended type 1 within type 241 would be "urn:ietf:gss:radius-attr
241.1".

The value of RADIUS attributes is the raw octets of the packet.
Integers are in network byte order. The display value SHOULD be a
human readable string; an implementation can only produce this string
if it knows the type of a given RADIUS attribute.

6. **Names of SAML Attributes in the Federated Context**

6.1. **Assertions**

   An assertion generated by the credential source is named by
   "urn:ietf:params:gss:fed-saml-assertion". The value of this
   attribute is the assertion carried in the AAA protocol or used for
   authentication in a SAML mechanism. This attribute is absent from a
   given acceptor name if no such assertion is present or if the
   assertion fails local policy checks. This attribute is always
   authentic when present: authentication only succeeds if the AAA
   exchange is successfully authenticated. However, users of the GSS-
   API MUST confirm that the attribute is authenticated because some
   mechanisms MAY permit an initiator to assert an unauthenticated
   version of this attribute.

6.2. **SAML Attributes**

   Each attribute carried in the assertion SHOULD also be a GSS name
   attribute. The name of this attribute has three parts, all separated
   by an ASCII space character. The first part is
   urn:ietf:params:gss:fed-saml-attr. The second part is the URI for
   the <saml:Attribute> element's NameFormat XML attribute. The final
   part is the <saml:Attribute> element's Name XML attribute.

   If the content of each <saml:AttributeValue> element is a simple text
   node (or nodes), then the raw and "display" values of the GSS name
   attribute MUST be the text content of the element(s). The raw value
   MUST be encoded as UTF-8.

   If the value is not simple, then the raw value(s) of the GSS name
   attribute MUST be the well-formed serialization of the <saml:
   AttributeValue> element(s) encoded as UTF-8. The "display" values
   are implementation-defined.

   These attributes SHOULD be marked authenticated if they are contained
   in SAML assertions that have been successfully validated back to the
   trusted source of the peer credential. In the GSS-EAP mechanism, a
   SAML assertion carried in an integrity-protected and authenticated
   AAA protocol SHALL be sufficiently validated. An implementation MAY
   apply local policy checks to this assertion and discard it if it is
   unacceptable according to these checks.

6.3. **SAML Name Identifiers**

   The <saml:NameID> carried in the subject of the assertion SHOULD also
   be a GSS name attribute. The name of this attribute has two parts,
   separated by an ASCII space character. The first part is

urn:ietf:params:gss:fed-saml-nameid. The second part is the URI for
the <saml:NameID> element's Format XML attribute.

The raw value of the GSS name attribute MUST be the well-formed
serialization of the <saml:NameID> element encoded as UTF-8. The
"display" value is implementation-defined. For formats defined by
section 8.3 of [SAMLCORE], missing values of the NameQualifier or
SPNameQualifier XML attributes MUST be populated in accordance with
the definition of the format prior to serialization. In other words,
the defaulting rules specified for the "persistent" and "transient"
formats MUST be applied prior to serialization.

This attribute SHOULD be marked authenticated if the name identifier
is contained in a SAML assertion that has been successfully validated
back to the trusted source of the peer credential. In the GSS-EAP
mechanism, a SAML assertion carried in an integrity-protected and
authenticated AAA protocol SHALL be sufficiently validated. An
implementation MAY apply local policy checks to this assertion and
discard it if it is unacceptable according to these checks.

7. **Security Considerations**

   This document describes how to access RADIUS attributes, SAML
   attributes and SAML assertions from some GSS-API mechanisms. These
   attributes are typically used for one of two purposes. The least
   sensitive is personalization: a central service MAY provide
   information about an authenticated user so they need not enter it
   with each acceptor they access. A more sensitive use is
   authorization.

   The mechanism is responsible for authentication and integrity
   protection of the attributes. However, the acceptor application is
   responsible for making a decision about whether the credential source
   is trusted to assert the attribute and validating the asserted value.

8. IANA Considerations

   First, a new registry needs to be created for GSS URNs. Then, this
   needs to be registered in the IETF's URN registry. Then this
   registry needs to be populated with URN items from this spec.

9. Acknowledgements

   Scott Cantor contributed significant text and multiple reviews of
   this document.

10. References

10.1. Normative References

   [I-D.ietf-abfab-gss-eap]
             Hartman, S. and J. Howlett, "A GSS-API Mechanism for the
             Extensible Authentication Protocol",
             draft-ietf-abfab-gss-eap-05 (work in progress),
             March 2012.

   [I-D.ietf-kitten-gssapi-naming-exts]
             Williams, N., Johansson, L., Hartman, S., and S.
             Josefsson, "GSS-API Naming Extensions",
             draft-ietf-kitten-gssapi-naming-exts-13 (work in
             progress), March 2012.

   [I-D.ietf-radext-radius-extensions]
             DeKok, A. and A. Lior, "Remote Authentication Dial In User
             Service (RADIUS) Protocol Extensions",
             draft-ietf-radext-radius-extensions-04 (work in progress),
             January 2012.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
             Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2743] Linn, J., "Generic Security Service Application Program
             Interface Version 2, Update 1", RFC 2743, January 2000.

10.2. Informative References

   [I-D.ietf-kitten-sasl-saml-ec]
             Cantor, S. and S. Josefsson, "SAML Enhanced Client SASL
             and GSS-API Mechanisms", draft-ietf-kitten-sasl-saml-ec-01
             (work in progress), February 2012.

Authors' Addresses

Sam Hartman
Painless Security

Email: hartmans-ietf@mit.edu

Josh Howlett
JANET(UK)

Email: josh.howlett@ja.net