ABFAB
Internet-Draft
Intended status: Informational
Expires: February 8, 2016

J. Howlett
Janet
S. Hartman
Painless Security
A. Perez-Mendez, Ed.
University of Murcia
August 7, 2015

A RADIUS Attribute, Binding, Profiles, Name Identifier Format, and Confirmation Methods for SAML
draft-ietf-abfab-aaa-saml-11

Abstract

This document describes the use of the Security Assertion Mark-up Language (SAML) with RADIUS in the context of the ABFAB architecture. It defines two RADIUS attributes, a SAML binding, a SAML name identifier format, two SAML profiles, and two SAML confirmation methods. The RADIUS attributes permit encapsulation of SAML assertions and protocol messages within RADIUS, allowing SAML entities to communicate using the binding. The two profiles describe the application of this binding for ABFAB authentication and assertion query/request, enabling a Relying Party to request authentication of, or assertions for, users or machines (Clients). These Clients may be named using a NAI name identifier format. Finally, the subject confirmation methods allow requests and queries to be issued for a previously authenticated user or machine without needing to explicitly identify them as the subject. These artifacts have been defined to permit application in AAA scenarios other than ABFAB, such as network access.

Table of Contents

1. Introduction

Within the ABFAB architecture [I-D.ietf-abfab-arch] it is often desirable to convey Security Assertion Mark-up Language (SAML) assertions and protocol messages.

SAML typically only considers the use of HTTP-based transports, known as bindings [OASIS.saml-bindings-2.0-os], which are primarily intended for use with the SAML V2.0 Web Browser Single Sign-On Profile [OASIS.saml-profiles-2.0-os]. However the goal of ABFAB is to extend the applicability of federated identity beyond the Web to other applications by building on the AAA framework. Consequently there exists a requirement for SAML to integrate with the AAA framework and protocols such as RADIUS [RFC2865] and Diameter [RFC3588], in addition to HTTP.

In summary this document specifies:

- Two RADIUS attributes to encapsulate SAML assertions and protocol messages respectively.

- A SAML RADIUS binding that defines how SAML assertions and protocol messages can be transported by RADIUS within a SAML exchange.

- A SAML name identifier format in the form of a Network Access Identifier.

- A profile of the SAML Authentication Request Protocol that uses the SAML RADIUS binding to effect SAML-based authentication and authorization.

- A profile of the SAML Assertion Query And Request Protocol that uses the SAML RADIUS binding to effect the query and request of SAML assertions.

- Two SAML Subject Confirmation Methods for indicating that a user or machine client is the subject of an assertion.

This document adheres to the guidelines stipulated by [OASIS.saml-bindings-2.0-os] and [OASIS.saml-profiles-2.0-os] for defining new SAML bindings and profiles respectively, and other conventions applied formally or otherwise within SAML. In particular where this document provides a 'Required Information' section for the binding and profiles that enumerate:

- A URI that uniquely identifies the protocol binding or profile.

- Postal or electronic contact information for the author.

- A reference to previously defined bindings or profiles that the new binding updates or obsoletes.

- In the case of a profile, any SAML confirmation method identifiers defined and/or utilized by the profile.

1.1. Terminology

This document uses terminology from a number of related standards, which tend to addopt different terms for similar or identical concepts. In general the document uses, when possible, the ABFAB term for the entity, as described in [I-D.ietf-abfab-arch]. For reference we include this table which maps the different terms into a single view.

Protocol	Client	Relying Party	Identity Provider
ABFAB	Client	Relying Party	Identity Provider
SAML	Subject Principal	Service Provider Requester Consumer	Identity Provider Responder Issuer
RADIUS	User	NAS RADIUS client	AS RADIUS server

Table 1. Terminology

2. Conventions

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 RFC 2119 [RFC2119].

3. RADIUS SAML Attributes

The RADIUS SAML binding defined in Section 4 of this document uses two attributes to convey SAML assertions and protocol messages respectively [OASIS.saml-core-2.0-os]. Owing to the typical size of these structures, these attributes use the Long Extended Type format [RFC6929] to encapsulate their data.

3.1. SAML-Assertion attribute

This attribute is used to encode a SAML assertion. The following figure represents the format of this attribute.

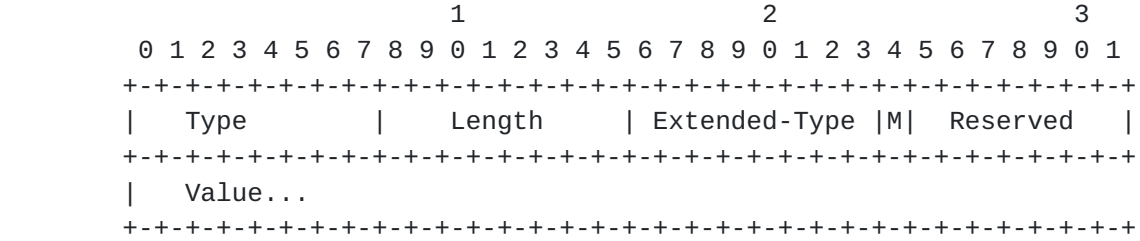


Figure 1: SAML-Assertion format

Type

245 (To be confirmed by IANA)

Length

>= 5

Extended-Type

TBD1

M (More)

As described in [RFC6929].

Reserved

As described in [RFC6929].

Value

One or more octets encoding a SAML assertion.

3.2. SAML-Message attribute

This attribute is used to encode a SAML protocol message. The following figure represents the format of this attribute.

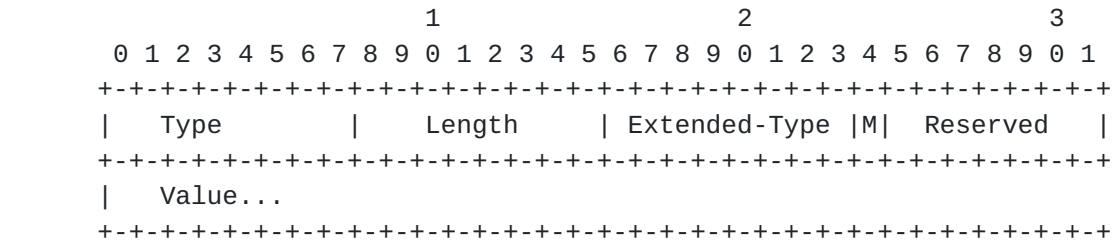


Figure 2: SAML-Message format

Type

245 (To be confirmed by IANA)

Length

>= 5

Extended-Type

TBD2

M (More)

As described in [RFC6929].

Reserved

As described in [RFC6929].

Value

One or more octets encoding a SAML protocol message.

4. SAML RADIUS Binding

The SAML RADIUS binding defines how RADIUS [RFC2865] can be used to enable a RADIUS client and server to exchange SAML assertions and protocol messages.

4.1. Required Information

Identification: urn:ietf:params:abfab:bindings:radius

Contact information: iesg@ietf.org

Updates: None.

4.2. Operation

RADIUS can be used over multiple underlying transports. It is REQUIRED that the RADIUS exchange is protected using TLS encryption for RADIUS [RFC6614] to provide confidentiality and improve integrity protection, unless alternative methods are used to ensure confidentiality, such as IPSEC tunnels or a sufficiently secure internal network.

Implementations of this profile can take advantage of mechanisms to permit the transport of longer SAML messages over RADIUS transports, such as the Support of fragmentation of RADIUS packets [RFC7499] or Larger Packets for RADIUS over TCP [I-D.ietf-radext-bigger-packets].

There are two system models for the use of SAML over RADIUS. The first is a request-response model, using the RADIUS SAML-Message attribute defined in Section 3 to encapsulate the SAML protocol messages.

1. The RADIUS client, acting as a Relying Party (RP), transmits a SAML request element within a RADIUS Access-Request message. This message MUST include a single instance of the RADIUS User-Name attribute whose value MUST conform to the Network Access

Identifier [RFC7542] scheme. The Relying Party MUST NOT include more than one SAML request element.

2. The RADIUS server, acting as an Identity Provider (IdP), returns a SAML protocol message within a RADIUS Access-Accept or Access-Reject message. These messages necessarily conclude a RADIUS exchange and therefore this is the only opportunity for the Identity Provider to send a response in the context of this exchange. The Identity Provider MUST NOT include more than one SAML response. An IdP that refuses to perform a message exchange with the Relying Party can silently discard the SAML request (this could subsequently be followed by a RADIUS Access-Reject, as the same conditions that cause the IdP to discard the SAML request may also cause the RADIUS server to fail to authenticate).

The second system model permits a RADIUS server acting as an Identity Provider to use the RADIUS SAML-Assertion attribute defined in Section 3 to encapsulate an unsolicited SAML assertion. This attribute MUST be included in a RADIUS Access-Accept message. When included, the attribute MUST contain a single SAML assertion.

RADIUS servers MUST NOT include both the SAML-Message and the SAML-Assertion attribute in the same RADIUS message. If an IdP is producing a response to a SAML request, then the first system model is used. An IdP MAY ignore a SAML request and send an unsolicited assertion using the second system model using the RADIUS SAML-Assertion attribute.

In either system model, Identity Providers SHOULD return a RADIUS state attribute as part of the Access-Accept message so that future SAML queries or requests can be run against the same context of an authentication exchange.

This binding is intended to be composed with other uses of RADIUS, such as network access. Therefore, other arbitrary RADIUS attributes will be used in either the request or response.

In the case of a SAML processing error and successful authentication, the RADIUS server SHOULD include a SAML-specified <samlp:Status> element in the SAML response that is transported within the Access-Accept packet sent by the RADIUS server.

In the case of a SAML processing error and failed authentication, the RADIUS server MAY include a SAML-specified <samlp:Status> element in the SAML response that is transported within the Access-Reject packet sent by the RADIUS server.

4.3. Processing of names

SAML entities using profiles of this binding will typically possess both the SAML and AAA names of their correspondents. Frequently these entities will need to apply policies using these names; for example, when deciding to release attributes. Often these policies will be security-sensitive, and so it is important that policy is applied on these names consistently.

4.3.1. AAA names

These rules relate to the processing of AAA names by SAML entities using profiles of this binding.

- Identity Providers SHOULD apply policy based on the Relying Party's identity associated with the RADIUS Access-Request.

- Relying Parties SHOULD apply policy based on the NAI realm associated with the RADIUS Access-Accept.

4.3.2. SAML names

These rules relate to the processing of SAML names by SAML entities using profiles of this binding.

Identity Providers MAY apply policy based on the Relying Party's SAML <entityId>. In such cases, at least one of the following methods is required in order to establish a relation between the SAML name and the AAA name of the Relying Party:

- RADIUS client identity in trusted digitally signed SAML request.

- RADIUS client identity in trusted SAML federation metadata (as in [OASIS.saml-metadata-2.0-os]).

A digitally signed SAML request without the RADIUS client identity is not sufficient, since a malicious RADIUS entity can observe a SAML message and include it in a different RADIUS message without the consent of the issuer of that SAML message. If an Identity Provider were to process the SAML message without confirming that it applied to the RADIUS message, inappropriate policy would be used.

Relying Parties MAY apply policy based on the SAML issuer's <entityId>. In such cases, at least one of the following methods is required in order to establish a relation between the SAML name and the AAA name of the Identity Provider:

- RADIUS realm in trusted digitally signed SAML response or assertion.

- RADIUS realm in trusted SAML federation metadata.

A digitally signed SAML response alone is not sufficient for the same reasons described above for SAML requests.

4.3.3. Mapping of AAA names in SAML metadata

This section defines the extensions to the SAML metadata specification [OASIS.saml-metadata-2.0-os] that are required in order to represent AAA names associated to a particular <EntityDescriptor> element.

In SAML metadata, each single entity may act in many different roles in the support of multiple profiles. This document defines two new roles: RADIUS IDP and RADIUS RP, requiring the declaration of two new subtypes of RoleDescriptorType: RADIUSIDPDescriptor and RADIUSRPDescriptor. These subtypes define the additional elements required to represent AAA names for IDP and RP entities respectively.

4.3.3.1. <RADIUSIDPDescriptor>

The <RADIUSIDPDescriptor> element extends RoleDescriptorType with elements common to IdPs that support RADIUS. Its RADIUSIDPDescriptorType complex type contains the following additional elements:

<RADIUSIDPService> [Zero or More] Zero or more elements of type EndpointType that describe RADIUS endpoints that are associated to this Entity.

<RADIUSRealm> [Zero or More] Zero or more elements of type xs:string that represent the acceptable values of the RADIUS realm associated to this Entity, obtained from the realm part of RADIUS User-Name attribute.

The following schema fragment defines the <RADIUSIDPDescriptor> element and its RADIUSIDPDescriptorType complex type:

```
    <element name="RADIUSIDPDescriptor"
             type="md:RADIUSIDPDescriptorType"/>
        <complexType name="RADIUSIDPDescriptorType">
            <complexContent>
                <extension base="md:RoleDescriptorType">
                    <sequence>
                        <element ref="md:RADIUSIDPService"
                                 minOccurs="0" maxOccurs="unbounded"/>
                        <element ref="md:RADIUSRealm"
                                 minOccurs="0" maxOccurs="unbounded"/>
                    </sequence>
                </extension>
            </complexContent>
        </complexType>
    <element name="RADIUSIDPService" type="md:EndpointType"/>
    <element name="RADIUSRealm" type="xs:string"/>
```

Figure 3: RADIUSIDPDescriptor schema

4.3.3.2. <RADIUSRPDescriptor>

The <RADIUSRPDescriptor> element extends RoleDescriptorType with elements common to RPs that support RADIUS. Its RADIUSRPDescriptorType complex type contains the following additional elements:

<RADIUSRPService> [Zero or More] Zero or more elements of type EndpointType that describe RADIUS endpoints that are associated to this Entity.

<RADIUSNasIpAddress> [Zero or More] Zero or more elements of type xs:string that represent the acceptable values of the RADIUS NAS-IP-Address attribute associated to this Entity.

<RADIUSNasIdentifier> [Zero or More] Zero or more elements of type xs:string that represent the acceptable values of the RADIUS NAS-Identifier attribute associated to this Entity.

<RADIUSGssEapName> [Zero or More] Zero or more elements of type xs:string that represent the acceptable values of the GSS-EAP acceptor name associated to this Entity. The format for this name is described in section 3.1 of [RFC7055], while section 3.4 describes how that name is decomposed and transported using RADIUS attributes.

The following schema fragment defines the <RADIUSRPDescriptor> element and its RADIUSRPDescriptorType complex type:

```
<element name="RADIUSRPDescriptor"
         type="md:RADIUSRPDescriptorType"/>
    <complexType name="RADIUSRPDescriptorType">
        <complexContent>
            <extension base="md:RoleDescriptorType">
                <sequence>
                    <element ref="md:RADIUSRPService"
                             minOccurs="0" maxOccurs="unbounded"/>
                    <element ref="md:RADIUSNasIpAddress"
                             minOccurs="0" maxOccurs="unbounded"/>
                    <element ref="md:RADIUSNasIdentifier"
                             minOccurs="0" maxOccurs="unbounded"/>
                    <element ref="md:RADIUSGssEapName"
                             minOccurs="0" maxOccurs="unbounded"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
<element name="RADIUSRPService" type="md:EndpointType"/>
<element name="RADIUSNasIpAddress" type="xs:string"/>
<element name="RADIUSNasIdentifier" type="xs:string"/>
<element name="RADIUSGssEapName" type="xs:string"/>
```

Figure 4: RADIUSRPDescriptor schema

4.3.4. Example of SAML metadata including AAA names

The following figures illustrate an example of metadata including AAA names for and IDP and a RP respectively. The IDP's SAML name is "https://IdentityProvider.com/", whereas its RADIUS realm is "idp.com". The RP's SAML name is "https://RelyingParty.com/SAML", being its GSS-EAP acceptor name "nfs/fileserver.rp.com@RP.COM".

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
                  entityID="https://IdentityProvider.com/SAML">
    <RADIUSIDPDescriptor protocolSupportEnumeration=
                         "urn:oasis:names:tc:SAML:2.0:protocol">
        <RADIUSRealm>
            idp.com
        </RADIUSRealm>
    </RADIUSIDPDescriptor>
</EntityDescriptor>
```

Figure 5: Metadata for the IDP

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
                  entityID="https://RelyingParty.com/SAML">
    <RADIUSRPDescriptor protocolSupportEnumeration=
                          "urn:oasis:names:tc:SAML:2.0:protocol">
        <RADIUSGssEapName>
            nfs/fileserver.rp.com@RP.COM
        </RADIUSGssEapName>
    </RADIUSRPDescriptor>
</EntityDescriptor>
```

Figure 6: Metadata for the RP

4.4. Use of XML Signatures

This binding calls for the use of SAML elements that support XML signatures. To promote interoperability, implementations of this binding MUST support a default configuration that does not require the use of XML signatures. Implementations MAY choose to use XML signatures.

4.5. Metadata Considerations

There are no metadata considerations particular to this binding, because this binding and profiles of this binding are mostly intended to be used without metadata. In this usage, RADIUS infrastructure is used to provide integrity and naming of the SAML messages and assertions. RADIUS configuration is used to provide policy, including which attributes are accepted from a Relying Party and which attributes are sent by an Identity Provider.

Nevertheless, implementations MAY support other configurations including the use of metadata.

5. Network Access Identifier Name Identifier Format

URI: urn:ietf:params:abfab:nameid-format:nai

Indicates that the content of the element is in the form of a Network Access Identifier (NAI) using the syntax described by [RFC7542].

6. RADIUS State Confirmation Method Identifiers

URI: urn:ietf:params:abfab:cm:user

URI: urn:ietf:params:abfab:cm:machine

Indicates that the Subject is the system entity (either the user or machine) authenticated by a previously transmitted RADIUS Access-

Accept message, as identified by the value of that RADIUS message's State attribute.

7. ABFAB Authentication Profile

In the scenario supported by the ABFAB Authentication Profile, a Client controlling a User Agent requests access to a Relying Party. The Relying Party uses RADIUS to authenticate the Client. In particular, the Relying Party, acting as a RADIUS client, attempts to validate the Client's credentials against a RADIUS server acting as the Client's Identity Provider. If the Identity Provider successfully authenticates the Client, it produces an authentication assertion which is consumed by the Relying Party. This assertion MAY include a name identifier that can be used between the Relying Party and the Identity Provider to refer to the Client.

7.1. Required Information

Identification: urn:ietf:params:abfab:profiles:authentication

Contact information: iesg@ietf.org

SAML Confirmation Method Identifiers: The SAML V2.0 "RADIUS State" confirmation method identifiers, either urn:ietf:params:abfab:cm:user or urn:ietf:params:abfab:cm:machine, are used by this profile.

Updates: None.

7.2. Profile Overview

To implement this scenario, a profile of the SAML Authentication Request protocol is used in conjunction with the SAML RADIUS binding defined in Section 4.

This profile is based on the SAML V2.0 Web Browser Single Sign-On Profile [OASIS.saml-profiles-2.0-os]. There are some important differences, specifically:

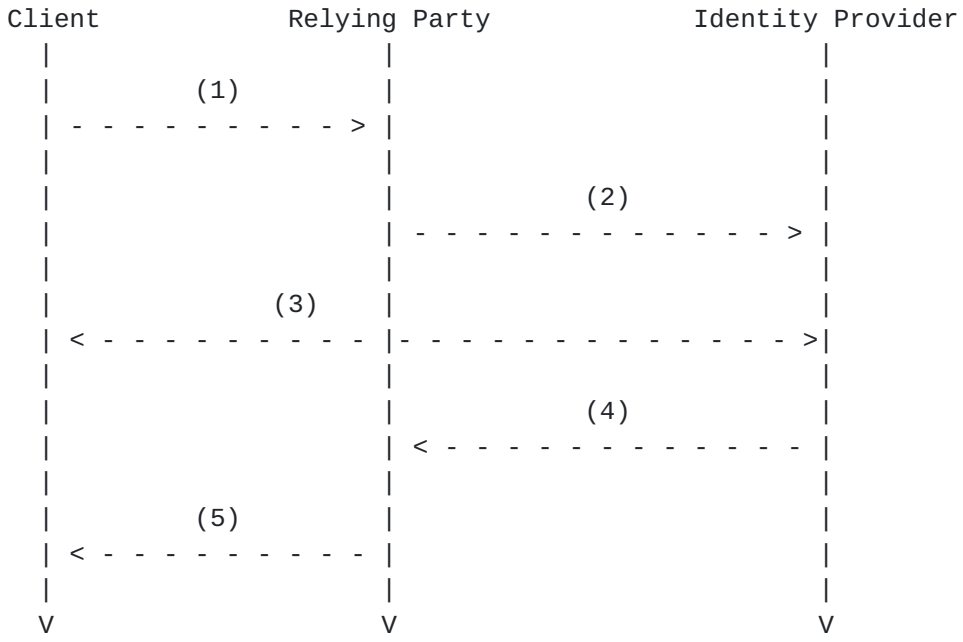
Authentication: This profile does not require the use of any particular authentication method. The ABFAB architecture does require the use of EAP [RFC3579], but this specification may be used in other non-ABFAB scenarios.

Bindings: This profile does not use HTTP-based bindings. Instead all SAML protocol messages are transported using the SAML RADIUS binding defined in Section 4. This is intended to reduce the number of bindings that implementations must support to be interoperable.

Requests: The profile does not permit the Relying Party to name the <saml:Subject> of the <samlp:AuthnRequest>. This is intended to simplify implementation and interoperability.

Responses: The profile only permits the Identity Provider to return a single SAML message or assertion that MUST contain exactly one authentication statement. Other statements may be included within this assertion at the discretion of the Identity Provider. This is intended to simplify implementation and interoperability.

Figure 7 below illustrates the flow of messages within this profile.



The following steps are described by the profile. Within an individual step, there may be one or more actual message exchanges.

Figure 7

1. Client request to Relying Party (Section 7.3.1): In step 1, the Client, via a User Agent, makes a request for a secured resource at the Relying Party. The Relying Party determines that no security context for the Client exists and initiates the authentication process.

2. Relying Party issues <samlp:AuthnRequest> to Identity Provider (Section 7.3.2). In step 2, the Relying Party may optionally issue a <samlp:AuthnRequest> message to be delivered to the Identity Provider using the SAML-Message RADIUS attribute.

3. Identity Provider identifies Client (Section 7.3.3). In step 3, the Client is authenticated and identified by the Identity Provider, while honoring any requirements imposed by the Relying Party in the <samlp:AuthnRequest> message if provided.

4. Identity Provider issues <samlp:Response> to Relying Party (Section 7.3.4). In step 4, the Identity Provider issues a <samlp:Response> message to the Relying Party using the SAML RADIUS binding. The response either indicates an error or includes a SAML Authentication Statement in exactly one SAML Assertion.

5. Relying Party grants or denies access to Client (Section 7.3.5). In step 5, having received the response from the Identity Provider, the Relying Party can respond to the Client with its own error, or can establish its own security context for the Client and return the requested resource.

7.3. Profile Description

The ABFAB Authentication Profile is a profile of the SAML V2.0 Authentication Request Protocol [OASIS.saml-core-2.0-os]. Where this specification conflicts with it, the former takes precedence.

7.3.1. Client Request to Relying Party

The profile is initiated by an arbitrary Client request to the Relying Party. There are no restrictions on the form of the request. The Relying Party is free to use any means it wishes to associate the subsequent interactions with the original request. The Relying Party, acting as a RADIUS client, attempts to authenticate the Client.

7.3.2. Relying Party Issues <samlp:AuthnRequest> to Identity Provider

The Relying Party uses RADIUS to communicate with the Client's Identity Provider. The Relying Party MAY include a <samlp:AuthnRequest> within this RADIUS Access-Request message using the SAML-Message RADIUS attribute. The next hop destination MAY be the Identity Provider or alternatively an intermediate RADIUS proxy.

Profile-specific rules for the contents of the <samlp:AuthnRequest> element are given in Section 7.4.1.

7.3.3. Identity Provider Identifies Client

The Identity Provider MUST establish the identity of the Client using a RADIUS authentication method, or else it will return an error. If the ForceAuthn attribute on the <samlp:AuthnRequest> element (if sent by the Relying Party) is present and true, the Identity Provider MUST freshly establish this identity rather than relying on any existing session state it may have with the Client (for example, TLS state that may be used for session resumption). Otherwise, and in all other respects, the Identity Provider may use any method to authenticate the Client, subject to the constraints called out in the <samlp:AuthnRequest> message.

7.3.4. Identity Provider Issues <samlp:Response> to Relying Party

The Identity Provider MUST conclude the authentication in a manner consistent with the RADIUS authentication result. The IdP MAY issue a <samlp:Response> message to the Relying Party that is consistent with the authentication result, as described in [OASIS.saml-core-2.0-os]. This SAML response is delivered to the Relying Party using the SAML RADIUS binding described in Section 4.

Profile-specific rules regarding the contents of the <samlp:Response> element are given in Section 7.4.2.

7.3.5. Relying Party Grants or Denies Access to Client

If issued by the Identity Provider, the Relying Party MUST process the <samlp:Response> message and any enclosed <saml:Assertion> elements as described in [OASIS.saml-core-2.0-os]. Any subsequent use of the <saml:Assertion> elements is at the discretion of the Relying Party, subject to any restrictions contained within the assertions themselves or from any previously established out-of-band policy that governs the interaction between the Identity Provider and the Relying Party.

7.4. Use of Authentication Request Protocol

This profile is based on the Authentication Request Protocol defined in [OASIS.saml-core-2.0-os]. In the nomenclature of actors enumerated in section 3.4 of that document, the Relying Party is the requester, the User Agent is the attesting entity and the Client is the Requested Subject.

7.4.1. <samlp:AuthnRequest> Usage

The Relying Party MUST NOT include a <saml:Subject> element in the request. The authenticated RADIUS identity identifies the Client to the Identity Provider.

A Relying Party MAY include any message content described in [OASIS.saml-core-2.0-os], section 3.4.1. All processing rules are as defined in [OASIS.saml-core-2.0-os].

If the Relying Party wishes to permit the Identity Provider to establish a new identifier for the Client if none exists, it MUST include a <saml:NameIDPolicy> element with the AllowCreate attribute set to "true". Otherwise, only a Client for whom the Identity Provider has previously established an identifier usable by the Relying Party can be authenticated successfully.

The <samlp:AuthnRequest> message MAY be signed. Authentication and integrity are also provided by the SAML RADIUS binding.

7.4.2. <samlp:Response> Message Usage

If the Identity Provider cannot or will not satisfy the request, it MUST either respond with a <samlp:Response> message containing an appropriate error status code or codes and/or respond with a RADIUS Access-Reject message.

If the Identity Provider wishes to return an error, it MUST NOT include any assertions in the <samlp:Response message>. Otherwise, if the request is successful (or if the response is not associated with a request), the <samlp:Response> element is subject conform to the following:

- It MAY be signed.

- It MUST contain exactly one <saml:Assertion>. The <saml:Subject> element of this assertion MUST refer to the authenticated RADIUS user.

- The assertion MUST contain a <saml:AuthnStatement>. Besides, the assertion MUST contain a <saml:Subject> element with at least one <saml:SubjectConfirmation> element containing a Method of urn:ietf:params:abfab:cm:user or urn:ietf:params:abfab:cm:machine that reflects the authentication of the Client to the Identity Provider. Since the containing message is in response to an <samlp:AuthnRequest>, the InResponseTo attribute MUST match the request's ID. The <saml:Subject> element MAY use the NAI Name

Identifier Format described in Section 5 to establish an identifier between the Relying Party and the IdP.

- Other conditions MAY be included as requested by the Relying Party or at the discretion of the Identity Provider. The Identity Provider is NOT obligated to honor the requested set of conditions in the <samlp:AuthnRequest>, if any.

7.4.3. <samlp:Response> Message Processing Rules

The Relying Party MUST do the following:

- Assume that the Client's identifier implied by a SAML <Subject> element, if present, takes precedence over an identifier implied by the RADIUS User-Name attribute.

- Verify that the InResponseTo attribute in the "RADIUS State" <saml:SubjectConfirmationData> equals the ID of its original <samlp:AuthnRequest> message, unless the response is unsolicited, in which case the attribute MUST NOT be present.

- If a <saml:AuthnStatement> used to establish a security context for the Client contains a SessionNotOnOrAfter attribute, the security context SHOULD be discarded once this time is reached, unless the Relying Party reestablishes the Client's identity by repeating the use of this profile.

- Verify that any assertions relied upon are valid according to processing rules in [OASIS.saml-core-2.0-os].

- Any assertion which is not valid, or whose subject confirmation requirements cannot be met MUST be discarded and MUST NOT be used to establish a security context for the Client.

7.4.4. Unsolicited Responses

An Identity Provider MAY initiate this profile by delivering an unsolicited <saml:Assertion> to a Relying Party. This MUST NOT contain any <saml:SubjectConfirmationData> elements containing an InResponseTo attribute.

7.4.5. Use of the SAML RADIUS Binding

It is RECOMMENDED that the RADIUS exchange is protected using TLS encryption for RADIUS [RFC6614] to provide confidentiality and improve integrity protection.

7.4.6. Use of XML Signatures

This profile calls for the use of SAML elements that support XML signatures. To promote interoperability implementations of this profile MUST NOT require the use of XML signatures. Implementations MAY choose to use XML signatures.

7.4.7. Metadata Considerations

There are no metadata considerations particular to this binding.

8. ABFAB Assertion Query/Request Profile

This profile builds on the SAML V2.0 Assertion Query/Request Profile defined by [OASIS.saml-profiles-2.0-os]. That profile describes the use of the Assertion Query and Request Protocol defined by section 3.3 of [OASIS.saml-core-2.0-os] with synchronous bindings, such as the SOAP binding defined in [OASIS.saml-bindings-2.0-os].

While the SAML V2.0 Assertion Query/Request Profile is independent of the underlying binding, it is nonetheless useful to describe the use of the SAML RADIUS binding defined in Section 4 of this document, in the interests of promoting interoperable implementations, particularly as the SAML V2.0 Assertion Query/Request Profile is most frequently discussed and implemented in the context of the SOAP binding.

8.1. Required Information

Identification: urn:ietf:params:abfab:profiles:query

Contact information: iesg@ietf.org

Description: Given below.

Updates: None.

8.2. Profile Overview

As with the SAML V2.0 Assertion Query/Request Profile defined by [OASIS.saml-profiles-2.0-os] the message exchange and basic processing rules that govern this profile are largely defined by Section 3.3 of [OASIS.saml-core-2.0-os] that defines the messages to be exchanged, in combination with the binding used to exchange the messages. The SAML RADIUS binding described in this document defines the binding of the message exchange to RADIUS. Unless specifically noted here, all requirements defined in those specifications apply.

Figure 8 below illustrates the basic template for the query/request profile.

The following steps are described by the profile.

Figure 8

1. Query/Request issued by Relying Party: In step 1, a Relying Party initiates the profile by sending an <AssertionIDRequest>, <SubjectQuery>, <AuthnQuery>, <AttributeQuery>, or <AuthzDecisionQuery> message to a SAML authority.

2. <Response> issued by SAML Authority: In step 2, the responding SAML authority (after processing the query or request) issues a <Response> message to the Relying Party.

8.3. Profile Description

8.3.1. Differences from the SAML V2.0 Assertion Query/Request Profile

This profile is identical to the SAML V2.0 Assertion Query/Request Profile, with the following exceptions:

- When processing the SAML request, the IdP MUST give precedence to the Client's identifier implied by RADIUS State attribute, if present, over the identifier implied by the SAML request's <Subject>, if any.

- In respect to sections 6.3.1 and 6.5 of [OASIS.saml-profiles-2.0-os], this profile does not consider the use of metadata (as in [OASIS.saml-metadata-2.0-os]). See Section 8.3.4.

- In respect to sections 6.3.2, 6.4.1, and 6.4.2 of [OASIS.saml-profiles-2.0-os], this profile additionally stipulates that implementations of this profile MUST NOT require the use of XML signatures. See Section 8.3.3.

8.3.2. Use of the SAML RADIUS Binding

The RADIUS Access-Request sent by the Relying Party:

- MUST include an instance of the RADIUS Service-Type attribute, having a value of Authorize-Only.

- SHOULD include the RADIUS State attribute, where this Query/ Request pertains to previously authenticated Client.

When processing the SAML request, the IdP MUST give precedence to the Client's identifier implied by RADIUS State attribute over the identifier implied by the SAML request's <Subject>, if any.

It is RECOMMENDED that the RADIUS exchange is protected using TLS encryption for RADIUS [RFC6614] to provide confidentiality and improve integrity protection.

8.3.3. Use of XML Signatures

This profile calls for the use of SAML elements that support XML signatures. To promote interoperability implementations of this profile MUST NOT require the use of XML signatures. Implementations MAY choose to use XML signatures.

8.3.4. Metadata Considerations

There are no metadata considerations particular to this binding.

9. Privacy considerations

The profiles defined in this document allow a Relying Party to request specific information about the Client, and allow an IdP to disclose information about that Client. In this sense, Identity Providers MUST apply policy to decide what information is released to a particular Relying Party. Moreover, the identity of the Client is typically hidden from the Relying Party unless informed by the Identity Provider. Conversely, the Relying Party does typically know the realm of the IdP, as it is required to being able to route the RADIUS packets to the right destination.

The kind of information that is released by the IdP MAY include generic attributes such as affiliation shared by many Clients. But even these generic attributes can help to identify a specific Client. Other kind of attributes MAY also provide a Relying Party with the ability to link the same Client between different sessions. Finally, other kind of attributes MAY provide a group of Relying Parties with

the ability to link the Client between them or with personally identifiable information about the Client.

These profiles do not directly provide a Client with a mechanism to express preferences about what information is released. That information can be expressed out-of-band, for example as part of the enrollment process.

The Relying Party MAY disclose privacy-sensitive information about itself as part of the request, although this is unlikely in typical deployments.

If RADIUS proxies are used and encryption is not used, the attributes disclosed by the IdP are visible to the proxies. This is a significant privacy exposure in some deployments. Ongoing work is exploring mechanisms for creating TLS connections directly between the RADIUS client and the RADIUS server to reduce this exposure. If proxies are used, the impact of exposing SAML assertions to the proxies needs to be carefully considered.

The use of TLS to provide confidentiality for the RADIUS exchange is strongly encouraged. Without this, passive eavesdroppers can observe the assertions.

10. Security Considerations

In this specification, the Relying Party MUST trust any statement in the SAML messages from the IdP in the same way that it trusts information contained in RADIUS attributes. These entities MUST trust the RADIUS infrastructure to provide integrity of the SAML messages.

Furthermore, the Relying Party MUST apply policy and filter the information based on what information the IdP is permitted to assert and on what trust is reasonable to place in proxies between them.

XML signatures and encryption are provided as an OPTIONAL mechanism for end-to-end security. These mechanism can protect SAML messages from being modified by proxies in the RADIUS infrastructure. These mechanisms are not mandatory-to-implement. It is believed that ongoing work to provide direct TLS connections between a RADIUS client and RADIUS server will provide similar assurances but better deployability. XML security is appropriate for deployments where end-to-end security is required but proxies cannot be removed or where SAML messages need to be verified at a later time or by parties not involved in the authentication exchange.

11. IANA Considerations

11.1. RADIUS Attributes

The authors request that Attribute Types and Attribute Values defined in this document be registered by the Internet Assigned Numbers Authority (IANA) from the RADIUS namespaces as described in the "IANA Considerations" section of [RFC3575], in accordance with BCP 26 [RFC5226]. For RADIUS packets, attributes and registries created by this document IANA is requested to place them at http://www.iana.org/assignments/radius-types.

In particular, this document defines two new RADIUS attributes, entitled "SAML-Assertion" and "SAML-Message" (see Section 3), with assigned values of 245.TBD1 and 245.TBD2 from the Long Extended Space of [RFC6929]:

Type	Ext. Type	Name	Length	Meaning
245	TBD1	SAML-Assertion	>=5	Encodes a SAML assertion
245	TBD2	SAML-Message	>=5	Encodes a SAML protocol message

11.2. ABFAB Parameters

A new top-level registry is created titled "ABFAB Parameters".

In this top-level registry, a sub-registry titled "ABFAB URN Parameters" is created. Registration in this registry is by the IETF review or expert review procedures [RFC5226].

This paragraph gives guidance to designated experts. Registrations in this registry are generally only expected as part of protocols published as RFCs on the IETF stream; other URIs are expected to be better choices for non-IETF work. Expert review is permitted mainly to allow early registration related to specifications under development when the community believes they have reached sufficient maturity. The expert SHOULD evaluate the maturity and stability of such an IETF-stream specification. Experts SHOULD review anything not from the IETF stream for consistency and consensus with current practice. Today such requests would not typically be approved.

If a parameter named "paramname" is to be registered in this registry, then its URN will be "urn:ietf:params:abfab:paramname". The initial registrations are as follows:

Parameter	Reference
bindings:radius	Section 4
nameid-format:nai	Section 5
profiles:authentication	Section 7
profiles:query	Section 8
cm:user	Section 6
cm:machine	Section 6

ABFAB Parameters

11.3. Registration of the ABFAB URN Namespace

IANA is requested to register the "abfab" URN sub-namespace in the IETF URN sub-namespace for protocol parameters defined in [RFC3553].

Registry Name: abfab

Specification: draft-ietf-abfab-aaa-saml

Repository: ABFAB URN Parameters (Section Section 11.2)

Index Value: Sub-parameters MUST be specified in UTF-8 using standard URI encoding where necessary.

12. Acknowledgements

The authors would like to acknowledge the OASIS Security Services (SAML) Technical Committee, and Scott Cantor in particular, for their help with the SAML-related material.

The authors would also like to acknowledge the collaboration of Jim Schaad, Leif Johansson, Klaas Wierenga, Stephen Farell, Gabriel Lopez, and Rafael Marin, who have provided valuable comments on this document.

13. References

13.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, DOI 10.17487/RFC2865, June 2000, <http://www.rfc-editor.org/info/rfc2865>.

[RFC3579] Aboba, B. and P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)", RFC 3579, DOI 10.17487/RFC3579, September 2003, <http://www.rfc-editor.org/info/rfc3579>.

[RFC6614] Winter, S., McCauley, M., Venaas, S., and K. Wierenga, "Transport Layer Security (TLS) Encryption for RADIUS", RFC 6614, DOI 10.17487/RFC6614, May 2012, <http://www.rfc-editor.org/info/rfc6614>.

[RFC6929] DeKok, A. and A. Lior, "Remote Authentication Dial In User Service (RADIUS) Protocol Extensions", RFC 6929, DOI 10.17487/RFC6929, April 2013, <http://www.rfc-editor.org/info/rfc6929>.

[RFC3575] Aboba, B., "IANA Considerations for RADIUS (Remote Authentication Dial In User Service)", RFC 3575, DOI 10.17487/RFC3575, July 2003, <http://www.rfc-editor.org/info/rfc3575>.

[RFC7542] DeKok, A., "The Network Access Identifier", RFC 7542, DOI 10.17487/RFC7542, May 2015, <http://www.rfc-editor.org/info/rfc7542>.

[OASIS.saml-bindings-2.0-os] Cantor, S., Hirsch, F., Kemp, J., Philpott, R., and E. Maler, "Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0", OASIS Standard saml-bindings-2.0-os, March 2005.

[OASIS.saml-core-2.0-os] Cantor, S., Kemp, J., Philpott, R., and E. Maler, "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0", OASIS Standard saml-core-2.0-os, March 2005.

[OASIS.saml-profiles-2.0-os] Hughes, J., Cantor, S., Hodges, J., Hirsch, F., Mishra, P., Philpott, R., and E. Maler, "Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0", OASIS Standard OASIS.saml-profiles-2.0-os, March 2005.

[OASIS.saml-metadata-2.0-os]
Cantor, S., Moreh, J., Philpott, R., and E. Maler, "Metadata for the Security Assertion Markup Language (SAML) V2.0", OASIS Standard saml-metadata-2.0-os, March 2005.

13.2. Informative References

[RFC3553] Mealling, M., Masinter, L., Hardie, T., and G. Klyne, "An IETF URN Sub-namespace for Registered Protocol Parameters", BCP 73, RFC 3553, DOI 10.17487/RFC3553, June 2003, <http://www.rfc-editor.org/info/rfc3553>.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, DOI 10.17487/RFC3588, September 2003, <http://www.rfc-editor.org/info/rfc3588>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[RFC7055] Hartman, S., Ed. and J. Howlett, "A GSS-API Mechanism for the Extensible Authentication Protocol", RFC 7055, DOI 10.17487/RFC7055, December 2013, <http://www.rfc-editor.org/info/rfc7055>.

[RFC7499] Perez-Mendez, A., Ed., Marin-Lopez, R., Pereniguez-Garcia, F., Lopez-Millan, G., Lopez, D., and A. DeKok, "Support of Fragmentation of RADIUS Packets", RFC 7499, DOI 10.17487/RFC7499, April 2015, <http://www.rfc-editor.org/info/rfc7499>.

[I-D.ietf-abfab-arch]
Howlett, J., Hartman, S., Tschofenig, H., Lear, E., and J. Schaad, "Application Bridging for Federated Access Beyond Web (ABFAB) Architecture", draft-ietf-abfab-arch-13 (work in progress), July 2014.

[I-D.ietf-radext-bigger-packets]
Hartman, S., "Larger Packets for RADIUS over TCP", draft-ietf-radext-bigger-packets-03 (work in progress), March 2015.

Authors' Addresses

Josh Howlett
Janet
Lumen House, Library Avenue, Harwell
Oxford OX11 0SG
UK

Phone: +44 1235 822363
EMail: Josh.Howlett@ja.net

Sam Hartman
Painless Security

EMail: hartmans-ietf@mit.edu

Alejandro Perez-Mendez (editor)
University of Murcia
Campus de Espinardo S/N, Faculty of Computer Science
Murcia 30100
Spain

Phone: +34 868 88 46 44
EMail: alex@um.es