A RADIUS Attribute, Binding, Profiles, Name Identifier Format, and Confirmation Methods for SAML
draft-ietf-abfab-aaa-saml-05

Abstract

This document specifies a RADIUS attribute, a binding, a name identifier format, two profiles, and two confirmation methods for the Security Assertion Mark-up Language (SAML). The attribute provides RADIUS encapsulation of SAML protocol messages, and the binding describes the use of this attribute, and the SAML protocol messages within, with RADIUS transport. The two profiles describe the application of this binding for ABFAB authentication and assertion query/request respectively. The name identifier format allows a subject to be named using an NAI, and the subject confirmation methods allow queries to be issued for a principal without needing to explicitly name the intended subject within the request. These artefacts have been defined to permit application in scenarios other than ABFAB, such as network access.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 29, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.
# Table of Contents

1. TODO ......................................................... 5
2. Introduction .................................................. 6
3. Conventions .................................................... 6
4. RADIUS SAML-Message Attribute ................................. 6
5. SAML RADIUS Binding ........................................... 7
   5.1. Required Information .................................... 7
   5.2. Operation ................................................ 7
       5.2.1. Use of XML Signatures ................................ 8
       5.2.2. Metadata Considerations ............................. 8
6. Network Access Identifier Name Identifier Format ................ 8
7. ABFAB Authentication Profile .................................. 8
   7.1. Required Information .................................... 9
   7.2. Profile Overview ......................................... 9
   7.3. Profile Description ....................................... 11
       7.3.1. User Agent Request to Relying Party ................ 11
       7.3.2. Relying Party Issues <samlp:AuthnRequest> to 
              Identity Provider .................................... 11
       7.3.3. Identity Provider Identifies Principal ............... 11
       7.3.4. Identity Provider Issues <samlp:Response> to 
              Relying Party ........................................ 12
       7.3.5. Relying Party Grants or Denies Access to Principal .. 12
   7.4. Use of Authentication Request Protocol .................... 12
       7.4.1. <samlp:AuthnRequest> Usage .......................... 12
       7.4.2. <samlp:Response message> Usage ...................... 13
       7.4.3. samlp:Response Message Processing Rules ............. 13
       7.4.4. Unsolicited Responses ................................ 14
       7.4.5. Use of the SAML RADIUS Binding ....................... 14
       7.4.6. Use of XML Signatures ................................ 14
       7.4.7. Metadata Considerations ............................. 14
8. ABFAB Assertion Query/Request Profile .......................... 14
   8.1. Required Information .................................... 15
   8.2. Profile Overview ......................................... 15
   8.3. Profile Description ....................................... 16
       8.3.1. Differences from the SAML V2.0 Assertion 
              Query/Request Profile ................................ 16
       8.3.2. Use of the SAML RADIUS Binding ....................... 16
       8.3.3. Use of XML Signatures ................................ 16
       8.3.4. Metadata Considerations ............................. 16
9. RADIUS State Confirmation Methods .............................. 16
10. Acknowledgements .............................................. 17
11. Security Considerations ....................................... 17
12. IANA Considerations ......................................... 17
   12.1. RADIUS Attributes ....................................... 17
   12.2. ABFAB Parameters ........................................ 17
   12.3. Registration of the ABFAB URN Namespace ................ 18
13. References ................................................... 18
13.1. Normative References .......................... 18
13.2. Informative References ......................... 19
1. TODO

- What NAI do we use in Access-Requests for Assertion Query/Requests? Possibly define well-known user@realm value for AA, etc? We probably also need a way to map NAI names to entityIDs.

- Clean up use of terminology (e.g., "principal") to ensure consistency with other ABFAB docs.

2. Introduction

The SAML RADIUS attribute, binding, profiles, name identifier format and confirmation methods are motivated by the requirements of the ABFAB architecture [I-D.ietf-abfab-arch]. In this architecture, it is often desirable to convey Security Assertion Mark-up Language (SAML) protocol messages between a SAML requester and SAML responder. This can be used, for example, to allow a Relying Party to request a SAML assertion from an Identity Provider that describes a particular principal.

SAML typically only considers the of use 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.

A companion specification [I-D.jones-diameter-abfab] specifies equivalent functionality for Diameter.

In summary this document specifies:

- A SAML RADIUS attribute that defines how to encapsulate a SAML protocol message within a RADIUS attribute.

- A SAML RADIUS binding that defines how SAML requesters and responders can exchange SAML protocol messages.

- An Authentication Profile that defines how the SAML RADIUS binding is used to effect SAML-based authentication and authorization.

- An Assertion Query/Request Profile that defines how the SAML RADIUS binding is used to effect SAML-based assertion request.
o Two subject confirmation methods for indicating that a user or machine principal respectively is the subject of an assertion.

This document aspires 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:

o A URI that uniquely identifies the protocol binding or profile

o Postal or electronic contact information for the author

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

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

3. 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].

4. RADIUS SAML-Message Attribute

This attribute contains a SAML [OASIS.saml-core-2.0-os] protocol message. Where multiple SAML-Message attributes are included in a RADIUS message, the Message fields of these attributes are to be concatenated to form a single SAML message.

A summary of the SAML-Message format is shown below. The fields are transmitted from left to right.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |         SAML Message...+
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1
5. SAML RADIUS Binding

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

5.1. Required Information

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

Contact information: iesg@ietf.org

Updates: None.

5.2. Operation

RADIUS can be used over multiple underlying transports; this binding calls out the use of Transport Layer Security (TLS) Encryption for RADIUS as REQUIRED [I-D.ietf-radext-radsec] to provide confidentiality, improve integrity protection and support the use of longer SAML messages.

Implementations of this profile MAY use RADIUS packet fragmentation [I-D.perez-radext-radius-fragmentation] to permit transport of longer SAML messages over RADIUS UDP transport.

The system model used for SAML conversations over RADIUS is a simple request-response model, using the RADIUS SAML-Message attribute defined in Section 4 to encapsulate the SAML protocol messages.

1. The RADIUS client, acting as a SAML requester, MAY transmit 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 [RFC4282] scheme. The SAML requester MUST NOT include more than one SAML request element.

2. The RADIUS server, acting as a SAML responder, MAY return a SAML protocol message within a RADIUS Access-Accept or Access-Reject message. If larger than a single attribute, the SAML message data MUST be split on 253-octet boundaries over as many attributes as necessary. The SAML message is reconstructed by concatenating the contents of all SAML-Message attributes.
message. These messages necessarily conclude a RADIUS exchange and therefore this is the only opportunity for the SAML responder to send a response in the context of this exchange. The SAML responder MUST NOT include more than one SAML response. A SAML responder that refuses to perform a message exchange with the SAML requester MUST silently discard the SAML request.

A SAML responder MAY also return an unsolicited response (a SAML response generated and emitted in the absence of a request from a SAML requester).

This binding is intended to be composed with other uses of RADIUS, such as network access. Therefore, other arbitrary RADIUS attributes MAY 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.

5.2.1. Use of XML Signatures

This bindings calls for the use of SAML elements that support XML signatures. To promote interoperability implementations of this binding MUST NOT require the use of XML signatures. Implementations MAY choose to use XML signatures, but this usage is outside of the scope of this binding.

5.2.2. Metadata Considerations

There are no metadata considerations particular to this binding.

6. 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 [RFC4282].

7. ABFAB Authentication Profile

In the scenario supported by the ABFAB Authentication Profile, a Principal controlling a User Agent requests access to a Relying
Party. The User Agent and Relying Party uses EAP to authenticate the Principal. The Relying Party, acting as an EAP pass-through authenticator, acts as a conduit for the EAP frames emitted by the User Agent and an EAP server which acts as the Principal’s Identity Provider. If the Identity Provider successfully authenticates the Principal, it produces an authentication assertion which is consumed by the Relying Party. During this process, a name identifier might also be established between the Relying Party and the Identity Provider.

7.1. Required Information

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

Contact information: iesg@ietf.org

SAML Confirmation Method Identifiers: The SAML V2.0 "sender vouches" confirmation method identifier, urn:oasis:names:tc:SAML:2.0:cm:sender-vouches, is 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 5, and EAP.

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 requires the use of EAP, although not a particular EAP authentication method. This allows the use of a variety of different EAP authentication mechanisms.

Bindings: This profile does not require the use of HTTP-based bindings. Instead all SAML protocol messages are transported using the SAML RADIUS binding defined in Section 5. 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 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 1 below illustrates the flow of messages within this profile.

```
<table>
<thead>
<tr>
<th>User Agent</th>
<th>Relying Party</th>
<th>Identity Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>(2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; ---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>(4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; ---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(5)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; ---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>V</td>
<td>V</td>
</tr>
</tbody>
</table>
```

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

**Figure 1**

1. User Agent Request to Relying Party (**Section 7.3.1**): In step 1, the Principal, 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 User Agent exists and initiates EAP authentication of the Principal.

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 RADIUS binding.

3. Identity Provider Identifies Principal (**Section 7.3.3**). In step 3, the Principal is identified by the Identity Provider using EAP authentication, 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 an authentication statement in exactly one assertion.

5. Relying Party Grants or Denies Access to Principal (Section 7.3.5). In step 5, having received the response from the Identity Provider, the Relying Party can respond to the Principal’s User Agent with its own error, or can establish its own security context for the Principal 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 Core, the former takes precedence.

7.3.1. User Agent Request to Relying Party

The profile is initiated by an arbitrary User Agent 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 an EAP authenticator, sends an EAP-Identity/Request message to the User Agent, acting as an EAP peer.

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

The Relying Party, on receiving the EAP-Identity/Response message from the User Agent, MUST send it towards the Identity Provider using RADIUS as described in [RFC3579]. The Relying Party MAY include a <samlp:AuthnRequest> within this RADIUS Access-Request message using the SAML RADIUS binding. 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 Principal

The Identity Provider MUST establish the identity of the Principal using EAP authentication, or else it will return an error. If the ForceAuthn attribute on the <samlp:AuthnRequest> element (if sent by the requester) 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 Principal (for example, TLS state
that may be used for session resumption). Otherwise, and in all other respects, the Identity Provider may use any EAP method to authenticate the Principal, subject to the requirements of Section 5.8 of [I-D.ietf-abfab-gss-eap] and any others called out in the <samlp:AuthnRequest> message.

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

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

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 Principal

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 on use contained within the assertions themselves or previously established out-of-band policy governing interactions 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, the Relying Party is the requester, the User Agent is the attesting entity and the Principal is the Requested Subject.

7.4.1. <samlp:AuthnRequest> Usage

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 Identity Provider cannot or will not satisfy the request, it MAY respond with a <samlp:Response> message containing an appropriate error status code or codes.

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

The Relying Party MUST NOT include a `<saml:Subject>` element in the request. The authenticated EAP Identity names the Principal of the requested `<samlp:AuthnRequest>` to the Identity Provider.

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

7.4.2. `<samlp:Response message>` Usage

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 MUST 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 Principal.
- The assertion MUST contain a `<saml:AuthnStatement>`. This MUST contain a `<saml:Subject>` element with at least one `<saml:SubjectConfirmation>` element containing a Method of urn:oasis:names:tc:SAML:2.0:cm:sender-vouches that reflects the authentication of the Principal to the Identity Provider. If the containing message is in response to an `<samlp:AuthnRequest>`, then the InResponseTo attribute MUST match the request’s ID.
- 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:

- Verify that the InResponseTo attribute in the sender-vouches `<samlp:SubjectConfirmationData>` equals the ID of its original `<samlp:AuthnRequest>` message, unless the response is unsolicited, in which case the attribute MUST NOT be present.
o If a <saml:AuthnStatement> used to establish a security context
   for the Principal contains a SessionNotOnOrAfter attribute, the
   security context SHOULD be discarded once this time is reached,
   unless the service provider reestablishes the Principal’s identity
   by repeating the use of this profile.

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

o 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 Principal.

7.4.4. Unsolicited Responses

An Identity Provider MAY initiate this profile by delivering an
unsolicited <samlp:Response> message to a Relying Party.

An unsolicited <samlp:Response> MUST NOT contain an InResponseTo
attribute, nor should any sender-vouches <saml:
SubjectConfirmationData> elements contain one.

7.4.5. Use of the SAML RADIUS Binding

It is RECOMMENDED that the RADIUS exchange is protected using TLS
encryption for RADIUS [I-D.ietf-radext-radsec] 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, but this usage is outside of the
scope of this profile.

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] or the SAML
RADIUS binding defined elsewhere in this document.
While the SAML V2.0 Assertion Query/Request Profile is independent of the underlying binding, it is nonetheless useful to describe the use of this profile with the SAML RADIUS binding 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](https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=saml-core) 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](https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=saml-core) 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 2 below illustrates the basic template for the query/request profile.

```
SAML Requester                  SAML Authority
|                               |
| (1)                           |
| - - - - - - - - - - - - - - - - - - - - - - > |
| (2)                           |
| < - - - - - - - - - - - - - - - - - - - - - - |
| V                            |
| V                            |
```

The following steps are described by the profile.

Figure 2
1. Query/Request issued by SAML Requester: In step 1, a SAML requester 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 SAML requester.

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:

- In respect to section 6.3.1 and 6.5, 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, 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

It is RECOMMENDED that the RADIUS exchange is protected using TLS encryption for RADIUS [I-D.ietf-radext-radsec] 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, but this usage is outside of the scope of this profile.

8.3.4. Metadata Considerations

There are no metadata considerations particular to this binding.

9. RADIUS State Confirmation Methods

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

URI: urn:ietf:params:abfab:cm:machine
The RADIUS State Confirmation Methods indicate 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.

10. Acknowledgements

TODO: Need to acknowledge OASIS SSTC, UoMurcia, Scott and Steven.

11. Security Considerations

TODO

12. IANA Considerations

12.1. RADIUS Attributes

Assignments of additional enumerated values for the RADIUS attribute defined in this document are to be processed as described in [RFC3575], subject to the additional requirements of a published specification.

12.2. 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 permit early registration related to specifications under development when the community believes they have reach 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 the "paramname" parameter is registered in this registry then its URN will be "urn:ietf:params:abfab:paramname". The initial registrations are as follows:
ABFAB Parameters

12.3. 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 12.2)

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

13. References

13.1. Normative References


[13.2. Informative References]


[OASIS.saml-profiles-2.0-os] Hughes, J., Cantor, S.,


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

Phone:
EMail: hartmans-ietf@mit.edu