ABFAB
Internet-Draft
Intended status: Informational
Expires: August 18, 2014

J. Howlett
Janet
S. Hartman
Painless Security
February 14, 2014

A RADIUS Attribute, Binding, Profiles, Name Identifier Format, and Confirmation Methods for SAML draft-ietf-abfab-aaa-saml-09

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, user or machine principals. These principals may be named using an 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. TODO

- Clean up use of terminology (e.g., "principal") to ensure consistency with other ABFAB docs.
- Complete the Acknowledgements and Security and Privacy Considerations sections.

2. Introduction

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

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

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 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 principal 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:

- 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

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 Attributes

The RADIUS SAML binding defined by this binding Section 5 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. The table below defines these attributes. The Length of both of these attributes is >=5. The More and Reserved fields are handled as described in [RFC6929] and are not depicted in this table for simplicity.

Name	Type	Extended-Type	Value
SAML-Assertion	TBD	TBD	One or more octets encoding a SAML assertion
SAML-Message	TBD	TBD	One or more octets encoding a SAML protocol message

Table 1: RADIUS SAML attribute definitions

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 assertions and 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 for the use of Transport Layer Security (TLS) Encryption for RADIUS [RFC6614] as REQUIRED to provide interoperability, confidentiality, improve integrity protection and support the use of longer SAML messages.

Implementations of this profile can take advantage of other mechanisms such as RADIUS packet fragmentation [I-D.perez-radext-radius-fragmentation] to permit transport of longer SAML messages over UDP-based RADIUS transports, such as those described in [RFC2865] and [I-D.ietf-radext-dtls]. Support for fragmentation over UDP is not mandatory.

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 4 to encapsulate the SAML protocol messages.

1. The RADIUS client, acting as a SAML requester, 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 [I-D.ietf-radext-nai] scheme. The SAML requester MUST NOT include more than one SAML request element.

2. The RADIUS server, acting as a SAML responder, 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 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 can silently discard the SAML request (this could subsequently be followed by a RADIUS Access-Reject, as the same conditions that cause the SAML responder 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 a SAML responder to use the RADIUS SAML-Assertion attribute defined in Section 4 to encapsulate an unsolicited, unencrypted SAML assertion. This attribute MAY 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 a SAML responder is producing a response to a SAML request, then the first system model is used. A SAML responder 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, SAML responders 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 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.

5.3. 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 policy 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.

5.3.1. AAA names

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

- SAML responders SHOULD apply policy based on the NAS identity associated with the RADIUS Access-Request.

- SAML requesters SHOULD apply policy based on the NAI realm associated with the RADIUS Access-Accept.

5.3.2. SAML names

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

SAML issuers MAY apply policy based on the requester's <entityId> after validating that the request comes from the NAS. The following methods are sufficient:

- NAS identity in trusted digitally signed request.

- NAS identity in trusted SAML federation metadata.

A digitally signed request alone is not sufficient. A RADIUS entity can observe a SAML message and include it in a RADIUS message without the consent of the issuer of that SAML message. If a SAML consumer were to process the SAML message without confirming that it applied to the RADIUS message, inappropriate policy would be used.

SAML consumers MAY apply policy based on the SAML issuer's <entityId> after validating that the response comes from the RADIUS server. The following methods are sufficient:

- RADIUS realm in trusted digitally signed request.

- RADIUS realm in trusted SAML federation metadata.

A digitally signed request alone is not sufficient.

5.3.3. 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 support a default configuration that does 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.

5.3.4. Metadata Considerations

There are no metadata considerations particular to this binding, because this binding and profiles of this binding are intended to be used without metadata. In this usage, RADIUS infrastructure is used to provide integrity and naming. RADIUS configuration is used to provide policy including which attributes are accepted from a SAML responder and which attributes are sent by a SAML responder.

Implementations MAY support other configurations including the use of metadata.

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 [I-D.ietf-radext-nai].

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 RADIUS to authenticate the Principal. The Relying Party, acting as a NAS, attempts to validate the Principal's credentials against a RADIUS server acting 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 conjuction with the SAML RADIUS binding defined in Section 5.

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:

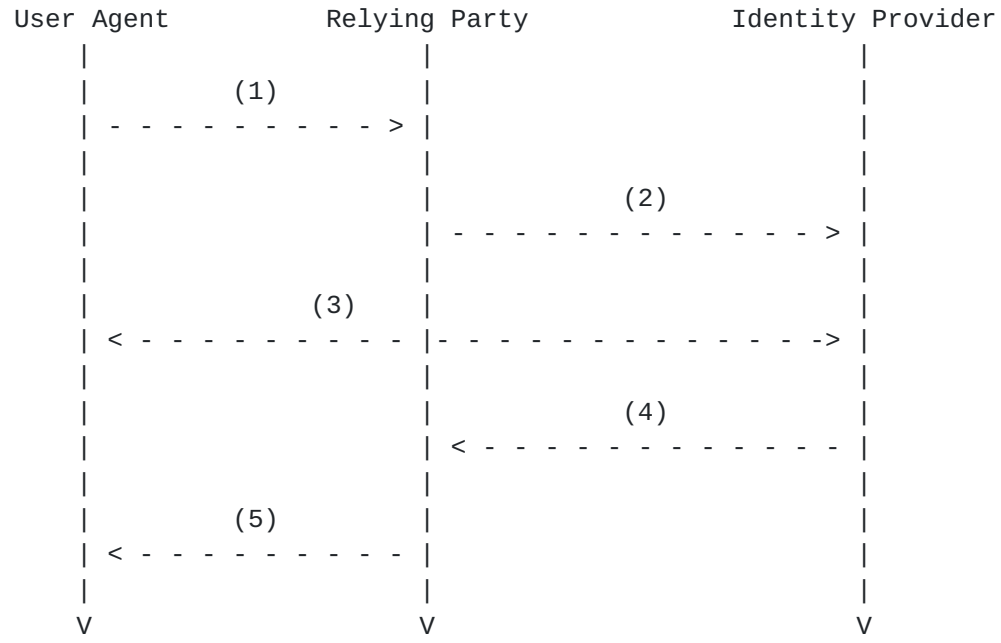
Authentication: This profile does not require the use of any particular authentication method. The ABFAB architecture does require the use of EAP [RFC3579], but this specification may be used in other non-ABFAB scenarios.

Bindings: This profile does not 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.



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 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 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 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 a NAS, attempts to authenticate the User Agent.

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

The Relying Party uses RADIUS to communicate with the Principal's Identity Provider. 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 RADIUS 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 method to authenticate the Principal, 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, and MAY issue a <samlp:Response> message to the Relying Party consisent 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

The Relying Party MUST NOT include a <saml:Subject> element in the request. The authenticated RADIUS user identifies the principal 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 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 <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 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 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 RADIUS user.
- 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:

- Assume that the principal implied by a SAML <Subject> element, if present, takes precedence over a principal implied by the RADIUS User-Name attribute.
- Verify that the InResponseTo attribute in the sender-vouches <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 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.

- 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 Principal.

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 sender-vouches <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, 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

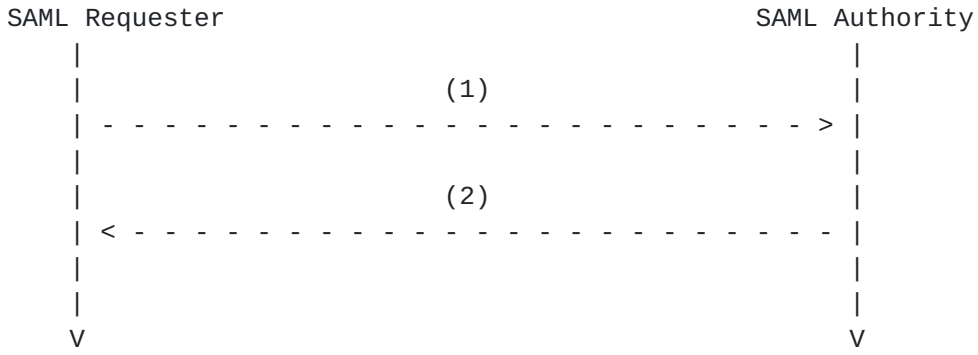
Description: Given below.

Updates: None.

8.2. Profile Overview

As with the SAML V2.0 Assertion Query/Request Profile defined by [OASIS.saml-profiles-2.0-os] 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 2 below illustrates the basic template for the query/request profile.



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:

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

- 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

The RADIUS Access-Request sent by the SAML requester:

- MUST use a RADIUS User-Name attribute whose value is "@REALM", where REALM is the destination NAI realm.

- 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 principal.

When processing the SAML request, the SAML responder MUST give precedence to the principal implied by RADIUS State attribute over the principal 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, 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, in the sense of [I-D.ietf-emu-eap-tunnel-method].

10. Privacy considerations

The profiles defined in this document allow a SAML requester to request specific information about the principal and allow a SAML responder to disclose information about a requester. Responders MUST apply policy to decide what information is released. The SAML requester does not typically know the identity of the principal unless informed by the SAML responder or RADIUS server. The SAML requester does typically know the realm of the IDP. Information that is released MAY include generic attributes such as affiliation shared by many principals. Even these generic attributes can help to identify a specific principal. Other attributes MAY provide a SAML requester with the ability to link the same principals between sessions with the same SAML requester. Other attributes MAY provide the requester with the ability to link the principal between requesters or with personally identifyable information about the principal.

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

The SAML requester MAY disclose privacy-sensitive information about

itself as part of the request. This is unlikely in typical deployments.

If RADIUS proxies are used, then attributes disclode by the SAML responder 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 NAS 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 observers can observe the assertions.

11. Acknowledgements

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

12. Security Considerations

TODO: Elaborate on the following

The RADIUS server vouches for its SAML messages. The NAS trusts any statement in the SAML messages from the RADIUS server in the same way that it trusts information contained in RADIUS attributes. The NAS MUST apply policy and filter the information based on what information the RADIUS server is permitted to assert and on what trust is reasonable to place in proxies between the NAS and RADIUS server.

SAML entities' level of trust in the SAML messages that they recieve from other entities should be consistent with the trust it holds in the RADIUS infrastructure. That is SAML entities SHOULD trust RADIUS to authenticate the principal and to reach the right IDP. SAML entities trust the RADIUS infrastructure to provide integrity of the SAML messages. However policy MUST be applied to limit what statements are permitted.

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 NAS 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 ro by parties not involved in the authentication exchange.

13. IANA Considerations

13.1. RADIUS Attributes

Assignments of additional enumerated values for the RADIUS attribute defined in this document are to be processed as described in [RFC6929], subject to the additional requirements of a published specification.

13.2. 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:

Parameter	Reference
bindings:radius	Section 5
nameid-format:nai	Section 6
profiles:authentication	Section 7
profiles:query	Section 8
cm:user	Section 9
cm:machine	Section 9

ABFAB Parameters

13.3. 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 13.2)

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

14. References

14.1. Normative References

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

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3579] Aboba, B. and P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)", RFC 3579, September 2003.

[RFC6614] Winter, S., McCauley, M., Venaas, S., and K. Wierenga, "Transport Layer Security (TLS) Encryption for RADIUS", RFC 6614, May 2012.

[RFC6929] DeKok, A. and A. Lior, "Remote Authentication Dial In User Service (RADIUS) Protocol Extensions", RFC 6929, April 2013.

[I-D.ietf-radext-nai] DeKok, A., "The Network Access Identifier", draft-ietf-radext-nai-03 (work in progress), May 2013.

[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.

14.2. 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, June 2003.

[RFC3575] Aboba, B., "IANA Considerations for RADIUS (Remote Authentication Dial In User Service)", RFC 3575, July 2003.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[I-D.perez-radext-radius-fragmentation] Perez-Mendez, A., Lopez, R., Pereniguez-Garcia, F., Lopez-Millan, G., Lopez, D., and A. DeKok, "Support of fragmentation of RADIUS packets", draft-perez-radext-radius-fragmentation-01 (work in progress), February 2012.

[I-D.jones-diameter-abfab] Jones, M. and H. Tschofenig, "The Diameter 'Application Bridging for Federated Access Beyond Web (ABFAB)' Application", draft-jones-diameter-abfab-00 (work in progress), March 2011.

[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-03 (work in progress), July 2012.

[I-D.ietf-radext-dtls] DeKok, A., "DTLS as a

Transport Layer for RADIUS", draft-ietf-radext-dtls-05 (work in progress), April 2013.

[I-D.ietf-emu-eap-tunnel-method] Zhou, H., Cam-Winget, N., Salowey, J., and S. Hanna, "Tunnel EAP Method (TEAP) Version 1", draft-ietf-emu-eap-tunnel-method-06 (work in progress), March 2013.

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