National Identity Exchange Federation

Cryptographic Trust Model

Version 1.1

July 31, 2018

Table of Contents

1. Target Audience and Purpose

This document specifies technical security and interoperability requirements for a National Identity Exchange Federation (NIEF) Cryptographic Trust Model. The purpose of this trust model is to provide a cryptographic foundation for secure communications and information sharing transactions within NIEF. All NIEF communication profiles, including the NIEF Web Browser User-to-System Profile [NIEF U2S], the NIEF Web Services System-to-System Profile [NIEF S2S], and the NIEF REST Services Profile [NIEF REST], rely on this trust model to provide a cryptographically secure basis for trusted communications between NIEF participants.

This document's target audience includes technical representatives of organizations who intend to participate in NIEF as Identity Provider Organizations (IDPOs), Service Provider Organizations (SPOs), Service Consumer Organizations (SCOs), Attribute Provider Organizations (APOs), or any allowable combination of these.[1] It also includes vendors, contractors, and consultants who, as part of their project or product implementation, have a requirement to establish technical interoperability with NIEF endpoints.

This document focuses only on issues of technical interoperability for the purpose of creating cryptographic trust. It does not cover governance, policy, or other nontechnical interoperability requirements. For more information about those topics, see [NIEF Bylaws] and [NIEF OPP].

2. NIEF Identity Trust Framework and Terminology

This document is one component of the NIEF Identity Trust Framework. See [NIEF OPP] for more information about the full NIEF Identity Trust Framework.

This document contains language that uses technical terms related to identity federations, identity management, and other related technologies. To minimize confusion for readers, it is important that each technical term have a precise definition. Accordingly, all technical terms in this document are to be interpreted as described in [NIEF Terms].

3. References

Table 1, Table 2, and Table 3 contain a list of documents that pertain to the specifications and requirements described in this document (including components from the NIEF Identity Assurance Framework and industry standards), and a list of reference URLs.

Document References for NIEF Identity Assurance Framework Components	
Document ID	**Document Name and URL if Applicable**

[1] See [NIEF Terms] for terminology related to various organizational and technical roles in NIEF.

NIEF Terms	NIEF Terminology Reference
NIEF Bylaws	NIEF Center Bylaws
NIEF OPP	NIEF Center Operational Policies and Procedures
NIEF Attrs	NIEF Attribute Registry
NIEF Attr Enc	NIEF Attribute Encoding Rules
NIEF CP	NIEF Certificate Policy
NIEF U2S	NIEF Web Browser User-to-System Profile
NIEF S2S	NIEF Web Services System-to-System Profile
NIEF REST	NIEF REST Services Profile

Table 1: Document References for NIEF Identity Assurance Framework Components

Document References for Industry Standards	
Document ID	**Document Name and URL**
SAML2 Core	"Assertions and Protocol for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-core-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf
SAML2 Bindings	"Bindings for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-bindings-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf
SAML2 Profiles	"Profiles for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-profiles-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf
SAML2 Metadata	"Metadata for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-metadata-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf
SAML2 Context	"Authentication Context for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-authn-context-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf
SAML2 Conform	"Conformance Requirements for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-conformance-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-conformance-2.0-os.pdf
SAML2 Security	"Security and Privacy Considerations for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-sec-consider-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-sec-consider-2.0-os.pdf
SAML2 Glossary	"Glossary for the OASIS Security Markup Language (SAML) V2.0" OASIS Standard, 15 March 2005 Document Identifier: saml-glossary-2.0-os http://docs.oasis-open.org/security/saml/v2.0/saml-glossary-2.0-os.pdf
SAML2 Assurance	"SAML V2.0 Identity Assurance Profiles, Version 1.0" OASIS Committee Specification 01, 5 November 2010 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-assurance-profile-cs-01.pdf
SAML2 Entity Attr	"SAML V2.0 Metadata Extension for Entity Attributes, Version 1.0" OASIS Committee Specification 01, 4 August 2009 http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.html

Document References for Industry Standards	
SAML2 Meta AR Ext	"SAML Metadata Extension for a Standalone Attribute Requester" Working Draft 01, 11 March 2005 Document Identifier: draft-saml-metadata-ext-01 https://www.oasis-open.org/committees/download.php/11805/draft-saml-metadata-ext-01.pdf
IDP Disc Profile	Identity Provider Discovery Service Protocol and Profile OASIS Committee Specification 01, 27 March 2008 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.pdf
FISMA	Federal Information Security Management Act http://csrc.nist.gov/sec-cert/
NIST SP 800-52	Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations National Institute of Science and Technology (NIST) Special Publication 800-52 http://csrc.nist.gov/publications/nistpubs/
NIST SP 800-63-2	Electronic Authentication Guideline National Institute of Science and Technology (NIST) Special Publication 800-63-2 http://csrc.nist.gov/publications/nistpubs/
OMB M-03-22	OMB Guidance for Implementing the Privacy Provisions of the E-Government Act of 2002 Office of Management and Budget (OMB) Memorandum M-03-22 http://www.whitehouse.gov/omb/memoranda/m03-22.html
RFC 2459	"RFC 2459— Internet X.509 Public Key Infrastructure Certificate and CRL Profile" Internet RFC/STD/FYI/BCP Archives http://tools.ietf.org/html/rfc2459
RFC 2119	"RFC 2119—Key Words for Use in RFCs to Indicate Requirement Levels" Internet RFC/STD/FYI/BCP Archives http://tools.ietf.org/html/rfc2119
FIPS 140-2	Federal Information Processing Standard (FIPS) Publication 140-2, "Security Requirements for Cryptographic Modules" http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf
OIDC Core	OpenID Connect Core 1.0 http://openid.net/specs/openid-connect-core-1_0.html
OIDC Disc	OpenID Connect Discovery 1.0 http://openid.net/specs/openid-connect-discovery-1_0.html
OIDC DCR	OpenID Connect Dynamic Client Registration 1.0 http://openid.net/specs/openid-connect-registration-1_0.html
OAuth DCR	OAuth 2.0 Dynamic Client Registration Protocol https://tools.ietf.org/html/rfc7591
JSON	RFC 7159 - The JavaScript Object Notation (JSON) Data Interchange Format http://tools.ietf.org/html/rfc7159
WebFinger	RFC 7033 - WebFinger http://tools.ietf.org/html/rfc7033
JWS	JSON Web Signature https://tools.ietf.org/html/rfc7515
JWK	JSON Web Key https://tools.ietf.org/html/rfc7517
JWT	JSON Web Token https://tools.ietf.org/html/rfc7519

Table 2: Document References for Industry Standards

Reference URLs for SAML and XML	
Topic	**Links**
SAML	http://www.oasis-open.org/home/index.php http://www.oasis-open.org/specs/index.php#samlv2.0 http://www.oasis-open.org/committees/security/ http://www.oasis-open.org/committees/security/docs
XML	http://www.w3.org/ http://www.w3.org/XML/ http://www.w3.org/1999/XMLSchema-instance http://www.w3.org/1999/XMLSchema

Table 3: Reference URLs

4. Notation

This document contains both normative and nonnormative content. Sections containing normative content are marked appropriately. In those sections, the key words "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" are to be interpreted as described in [RFC 2119].

5. NIEF Cryptographic Trust Model

All subsections that follow are normative, unless otherwise noted.

5.1 NIEF Cryptographic Trust Fabric Overview (Nonnormative)

The NIEF Cryptographic Trust Fabric defines the most current NIEF cryptographic security context and contains an entity descriptor entry for each communications endpoint in NIEF, including identity providers, relying parties, attribute providers, web service endpoints, and others. NIEF supports different types of communications protocols, including Security Assertion Markup Language (SAML)[2], Simple Object Access Protocol (SOAP), OpenID Connect (OIDC), and OAuth. NIEF considers OIDC and OAuth endpoints to be Representational State Transfer (REST) endpoints. NIEF SAML and SOAP endpoints have metadata conventions that use the SAML Metadata format. These conventions are different than those used by REST endpoints which use JavaScript Object Notation (JSON) formats. Therefore, the NIEF Cryptographic Trust Fabric is comprised of two Trust Fabric documents: the NIEF SAML Cryptographic Trust Fabric document (see Section 5.2), which contains trust metadata for SAML and SOAP communications endpoints, and the NIEF REST Cryptographic Trust Fabric document (see Section 5.3), which contains trust metadata for OIDC and OAuth communications endpoints.

The NIEF Center cryptographically signs the NIEF Cryptographic Trust Fabric documents. The NIEF Center does NOT issue certificates to NIEF participants, and although it is possible for the NIEF Center to sign NIEF participants' certificates, it is not

[2] The term "SAML communications endpoint" refers to an endpoint that speaks a SAML protocol, not necessarily to an endpoint that issues or consumes SAML assertions.

required. Trust between NIEF participants is anchored by the NIEF Cryptographic Trust Fabric documents and the NIEF Center's cryptographic signature of these documents.

The NIEF Center maintains the Cryptographic Trust Fabric documents and makes new versions of them available to NIEF participants periodically as needed due to the addition or removal of communications endpoints. To ensure compliance with the current NIEF Cryptographic Trust Fabric, each communications endpoint in NIEF MUST incorporate the most current version of the appropriate NIEF Cryptographic Trust Fabric documents into its security policy decisions in a timely fashion. The NIEF Center shall provide guidance to NIEF participants as to the urgency with which a new Trust Fabric document must be incorporated, at the time the new document is made available. In cases in which the new Trust Fabric document has been published because of a security or trust violation, or because of the removal of a member from NIEF for disciplinary reasons, it is recommended that members incorporate the new Trust Fabric document as soon as is reasonably possible, and not more than 24 hours after its release.

5.2 NIEF SAML Cryptographic Trust Fabric Document

The NIEF SAML Cryptographic Trust Fabric document contains trust metadata descriptors for SAML and SOAP communications endpoints and conforms to the SAML 2.0 standard for federated system entity metadata specification defined in [SAML2 Metadata]. It also uses several extension schemas,

1. A custom extension schema for the **<md:RoleDescriptor>** element serves to accommodate the use of NIEF Cryptographic Trust Fabric with various NIEF Web Services endpoints, by defining the types of NIEF Web Services endpoint elements that **<md:RoleDescriptor>** can contain. Appendix A contains this schema. It is also available at the following URL.

   https://www.gfipm.net/standards/metadata/2.1/webservices

2. A SAML standard extension schema for the **<mdattr:EntityAttributes>** element serves to accommodate the use of entity attributes, which are facts or properties asserted by the NIEF Center about a NIEF Cryptographic Trust Fabric entity and its endpoints. This schema is defined by [SAML2 Entity Attr].

3. Another SAML standard extension schema for the **<md:RoleDescriptor>** element defines a new element type, **AttributeRequesterDescriptorType**, that serves to enable a NIEF Cryptographic Trust Fabric to contain information about an endpoint that belongs to a SAML "Attribute Requester", i.e., a consumer of attributes from an Attribute Provider. This schema is defined by [SAML2 Meta AR Ext].

Additional constraints specified in this section also apply to the NIEF SAML Cryptographic Trust Fabric document.

5.2.1 SAML <EntitiesDescriptor> Element Requirements

The following additional requirements apply to the **<EntitiesDescriptor>** element, which is the top-level XML element within the NIEF SAML Cryptographic Trust Fabric document. These requirements supplement the requirements described in [SAML2 Metadata].

1. The **Name** attribute within the top-level **<EntitiesDescriptor>** MUST be present.

2. The **ID** attribute within the top-level **<EntitiesDescriptor>** MUST be present.

3. The **validUntil** and **cacheDuration** attributes within **<EntitiesDescriptor>** MUST be present. It is RECOMMENDED that **cacheDuration** not exceed 18 hours.

4. The **<ds:Signature>** element within the top-level **<EntitiesDescriptor>** MUST be present, and it MUST contain a valid signature for the document, signed by the NIEF Center.

5. The **<Extensions>** element within the top-level <**EntitiesDescriptor>** MUST NOT be present.

6. Nested **<EntitiesDescriptor>** elements within the top-level **<EntitiesDescriptor>** MAY be present. Nested **<EntitiesDescriptor>** elements SHOULD include **Name** and **ID** attributes.

7. One or more **<EntityDescriptor>** elements within **<EntitiesDescriptor>** MAY be present.

5.2.2 SAML <EntityDescriptor> Element Requirements

The following requirements apply to **<EntityDescriptor>** elements that appear in the NIEF SAML Cryptographic Trust Fabric document. Each **<EntityDescriptor>** element provides entity metadata for a specific entity that fulfills a NIEF Technical Role (see [NIEF Terms]). These requirements supplement the requirements described in [SAML2 Metadata].

1. The **entityID** attribute within **<EntityDescriptor>** MUST be present, MUST be unique within NIEF, and MUST be set to the value that was agreed upon for this entity between the entity and the NIEF Center. (The entity chooses its **entityID** value, but the choice MUST be approved by the NIEF Center.) Also, the requirements in the following subsections apply.

a. For an SP, the **entityID** attribute MUST be a URL that is under the SP's control.

b. For an AC, the **entityID** attribute MUST be a URN of the format "**urn:idmanagement.gov:icam:bae:v2:[LI]**" where "**[LI]**" is to be replaced with a value that was agreed upon between the AC and the NIEF Center.[3]

c. For an AP, the **entityID** attribute MUST be a URN of the format "**urn:idmanagement.gov:icam:bae:v2:[LI]**" where "**[LI]**" is to be replaced with a value that was agreed upon between the AP and the NIEF Center.[3]

2. If the **<EntityDescriptor>** appears within a signed **<EntitiesDescriptor>**, then the **<ds:Signature>** element within **<EntityDescriptor>** MUST NOT be present. If the **<EntityDescriptor>** is standalone, then the **<ds:Signature>** element MUST be present, and must contain a digital signature from the NIEF Center.

3. The **validUntil** and **cacheDuration** attributes within **<EntityDescriptor>** MUST be present, with their values set using risk-based methods. It is RECOMMENDED that **cacheDuration** not exceed 18 hours.

4. Each **<EntityDescriptor>** element MUST contain at least one **<IDPSSODescriptor>** element, OR at least one **<SPSSODescriptor>** element, OR at least one **<AttributeAuthorityDescriptor>** element, OR at least one **<RoleDescriptor>** element, and MAY contain additional **<IDPSSODescriptor>**, **<SPSSODescriptor>**, **<AttributeAuthorityDescriptor>**, and **<RoleDescriptor>** elements.

5. Each **<EntityDescriptor>** element MUST contain at least one **<ContactPerson>** element with each technical **contactType**. An **<EntityDescriptor>** element MAY contain additional **<ContactPerson>** elements.

6. The following requirements apply to each **<ContactPerson>** element within an **<EntityDescriptor>** element.

[3] The purpose of this requirement is to support the NIEF-WS Attribute Provider SIP.

a. The **<Extensions>** element MUST NOT be present.

b. The **<Company>** element MUST be present.

c. The **<GivenName>** element MUST be present.

d. The **<SurName>** element MUST be present.

e. At least one **<EmailAddress>** element is MUST be present.

f. At least one **<TelephoneNumber>** element MUST be present.

7. The **<AdditionalMetadataLocation>** element within **<EntityDescriptor>** MUST NOT be present.

8. Each **<EntityDescriptor>** element MAY contain one **<Extensions>** element, and the **<Extensions>** element MAY contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MAY contain zero or more SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

9. If the **<EntityDescriptor>** element pertains to an IDP that has been certified at one or more NIST Levels of Assurance under the Federal Identity, Credentialing, and Access Management (FICAM) trust framework or an equivalent trust framework, then it MUST contain one **<Extensions>** element, and the **<Extensions>** element MUST contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MUST contain a SAML **<Attribute>** element containing the list of NIST Levels of Assurance (LOAs) that the IDP is certified to assert. These LOAs MUST be expressed in accordance with the SAML Identity Assurance Certification Attribute Profile, which can be found in Section 3 of [SAML2 Assurance]. Specific **<AttributeValue>** elements MUST contain values that pertain to valid FICAM LOAs, as per the following list of values.

   - **http://idmanagement.gov/ns/assurance/loa/1**
   - **http://idmanagement.gov/ns/assurance/loa/2**
   - **http://idmanagement.gov/ns/assurance/loa/3**
   - **http://idmanagement.gov/ns/assurance/loa/4**

   In this case, the **<mdattr:EntityAttributes>** element MAY also contain zero or more additional SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

10. Each **<EntityDescriptor>** element SHOULD contain one **<Organization>** element. If it is present, the **<Organization>** element SHOULD contain at least one of each of these elements: **<OrganizationName>**, **<OrganizationDisplayName>**, and **<OrganizationURL>**.

5.2.3 SAML <IDPSSODescriptor> Element Requirements

The following requirements apply to **<IDPSSODescriptor>** elements that appear in the NIEF SAML Cryptographic Trust Fabric document. Each **<IDPSSODescriptor>** element provides metadata for the SAML services provided by a specific IDP. These requirements supplement the requirements described in [SAML2 Metadata].

1. The **protocolSupportEnumeration** attribute within **<IDPSSODescriptor>** MUST be present, and MUST have a value of "**urn:oasis:names:tc:SAML:2.0:protocol**".

2. The **WantAuthnRequestsSigned** attribute within **<IDPSSODescriptor>** MUST be present, and its value MUST be "**true**".

3. The **<ds:Signature>** element within **<IDPSSODescriptor>** MUST NOT be present.

4. Each **<IDPSSODescriptor>** element MAY contain one **<Extensions>** element, and the **<Extensions>** element MAY contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MAY contain zero or more SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

5. At least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**signing**" MUST be present within **<IDPSSODescriptor>**.

6. Each **<KeyDescriptor>** element MUST include a **<ds:KeyInfo>** element containing exactly one **<ds:X509Data>** element, and the **<ds:X509Data>** element MUST contain exactly one **<ds:X509Certificate>** element. Other sub-elements of the **<KeyDescriptor>** element are permitted, but they MUST represent the same key.

7. The **<ArtifactResolutionService>** element within **<IDPSSODescriptor>** MUST NOT be present.

8. The **<ManageNameIDService>** element within **<IDPSSODescriptor>** MUST NOT be present.

9. Two **<NameIDFormat>** elements MUST be present within **<IDPSSODescriptor>**. One MUST have a value of "**urn:oasis:names:tc:SAML:2.0:nameid-format:persistent**" and the other MUST have a value of "**urn:oasis:names:tc:SAML:2.0:nameid-format:transient**".

10. One **<SingleSignOnService>** element MUST be present within **<IDPSSODescriptor>**; its **Binding** attribute MUST be present, and the value of its **Binding** attribute MUST be "**urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect**". Also, its **Location** attribute MUST be present, with a value specifying the live service endpoint (URL) of this IDP's SAML HTTP Redirect Single Sign-On (SSO) service.

11. The **<IDPSSODescriptor>** element MUST contain a series of SAML **<Attribute>** elements, with one **<Attribute>** element for each attribute supported by the IDP.

12. The **<NameIDMappingService>** element within **<IDPSSODescriptor>** MUST NOT be present.

13. The <**AssertionIDRequestService>** element within **<IDPSSODescriptor>** MUST NOT be present.

14. The **<AttributeProfile>** element within **<IDPSSODescriptor>** MUST NOT be present.

5.2.4 SAML <SPSSODescriptor> Element Requirements

The following requirements apply to **<SPSSODescriptor>** elements that appear in the NIEF SAML Cryptographic Trust Fabric document. Each **<SPSSODescriptor>** element provides metadata for the SAML services provided by a specific NIEF SP. These requirements supplement the requirements described in [SAML2 Metadata].

1. The **protocolSupportEnumeration** attribute within **<SPSSODescriptor>** MUST be present, and its value MUST be "**urn:oasis:names:tc:SAML:2.0:protocol**".

2. The **WantAssertionsSigned** attribute MUST be present within **<SPSSODescriptor>**, and its value MUST be "**true**".

3. The **<ds:Signature>** element within **<SPSSODescriptor>** MUST NOT be present.

4. Each **<SPSSODescriptor>** element MAY contain one **<Extensions>** element, and the **<Extensions>** element MAY contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MAY contain zero or more SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

5. At least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**signing**" MUST be present within **<SPSSODescriptor>**.

6. At least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**encryption**" MUST be present within **<SPSSODescriptor>**.

7. Each **<KeyDescriptor>** element MUST include a **<ds:KeyInfo>** element containing exactly one **<ds:X509Data>** element, and the **<ds:X509Data>** element MUST contain exactly one **<ds:X509Certificate>** element. Other sub-elements of the **<KeyDescriptor>** element are permitted, but they MUST represent the same key.

8. The **<ArtifactResolutionService>** element within **<SPSSODescriptor>** MUST NOT be present.

9. The **<ManageNameIDService>** element within **<SPSSODescriptor>** MUST NOT be present.

10. At least one **<NameIDFormat>** element MUST be present within **<SPSSODescriptor>**, and its value MUST be "**urn:oasis:names:tc:SAML:2.0:nameid-format:persistent**" or "**urn:oasis:names:tc:SAML:2.0:nameid-format:transient**".

11. A second **<NameIDFormat>** element MAY be present within **<SPSSODescriptor>**. If present, its value MUST be "**urn:oasis:names:tc:SAML:2.0:nameid-format:persistent**" or "**urn:oasis:names:tc:SAML:2.0:nameid-format:transient**", and it MUST NOT have the same value as the

first **<NameIDFormat>** element. The maximum number of **<NameIDFormat>** elements allowed is two.

12. Exactly one **<AssertionConsumerService>** element MUST be present within **<SPSSODescriptor>**. Its **Binding** attribute MUST be present and MUST have a value of "**urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST**". Also, its **Location** attribute MUST be present, with a value specifying the live service endpoint (URL) of this SP's SAML HTTP POST Assertion Consumer Service.

13. If the SP requires specific user attributes (e.g. from [NIEF Attrs]) as a condition of access to any of its services, then one or more **<AttributeConsumingService>** elements within **<SPSSODescriptor>** MUST be present, and each **<AttributeConsumingService>** element MUST contain the list of required user attributes for access to that service. Required attributes MUST be indicated using the **<RequestedAttribute>** element. The SP MAY either publish one **<AttributeConsumingService>** element containing the user attribute requirements for all of its available services (resources), or a series of **<AttributeConsumingService>** elements, with one element per service (resource) that it offers.

5.2.5 SAML <AttributeAuthorityDescriptor> Element Requirements

The following requirements apply to **<AttributeAuthorityDescriptor>** elements that appear in the NIEF SAML Cryptographic Trust Fabric document. Each **<AttributeAuthorityDescriptor>** element provides metadata for the SAML services provided by a specific NIEF AP. These requirements supplement the requirements described in [SAML2 Metadata].

1. The **protocolSupportEnumeration** attribute within **<AttributeAuthorityDescriptor>** MUST be present, and MUST include both of the following values:

    - "**urn:oasis:names:tc:SAML:2.0:protocol**"
    - "**http://gfipm.net/standards/webservices/1.1/attribute-provider-sip.html**".

2. The **<ds:Signature>** element within **<AttributeAuthorityDescriptor>** MUST NOT be present.

3. Each **<AttributeAuthorityDescriptor>** element MAY contain one **<Extensions>** element, and the **<Extensions>** element MAY

contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MAY contain zero or more SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

4. At least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**signing**" MUST be present within **<AttributeAuthorityDescriptor>**.

5. If the AP endpoint supports XML encryption, then at least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**encryption**" MUST be present within **<AttributeAuthorityDescriptor>**.

6. Each **<KeyDescriptor>** element MUST include a **<ds:KeyInfo>** element containing exactly one **<ds:X509Data>** element, and the **<ds:X509Data>** element MUST contain exactly one **<ds:X509Certificate>** element. Other sub-elements of the **<KeyDescriptor>** element are permitted, but they MUST represent the same key.

7. At least one **<NameIDFormat>** element MUST be present within **<AttributeAuthorityDescriptor>**. The following additional requirements also apply.

    a. If the AP supports the use of the GFIPM Federation Id User Attribute for subject identification, then one of the **<NameIDFormat>** elements MUST have a value of "**urn:idmanagement.gov:icam:bae:v2:SAML:2.0:nameid-format:gfipm:2.0:user:FederationId**".

    b. If the AP supports the use of the GFIPM Email Address Text User Attribute for subject identification, then one of the **<NameIDFormat>** elements MUST have a value of "**urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress**".

    c. If the AP supports the use of the FASC-N identifier of a PIV credential for subject identification, then one of the **<NameIDFormat>** elements MUST have a value of "**urn:idmanagement.gov:icam:bae:v2:SAML:2.0:nameid-format:fasc-n**".

    d. If the AP supports the use of the UUID of a PIV-I credential for subject identification, then one of the **<NameIDFormat>**

elements MUST have a value of "**urn:idmanagement.gov:icam:bae:v2:SAML:2.0:nameid-format:uuid**".

e. If the AP supports the use of the X.509 Subject Distinguished Name for subject identification, then one of the **<NameIDFormat>** elements MUST have a value of "**urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName**".

8. At least one **<AttributeService>** element MUST be present within **<AttributeAuthorityDescriptor>**. The requirements in the following subsections apply to each **<AttributeService>** element.

   a. The **Binding** attribute MUST be present, and its value MUST be "**urn:oasis:names:tc:SAML:2.0:bindings:SOAP**".

   b. The **Location** attribute MUST be present and its value MUST specify a live service endpoint (URL) of the AP.

9. The **<AttributeAuthorityDescriptor>** element MUST contain a series of SAML **<Attribute>** elements, with one **<Attribute>** element for each attribute supported by the AP.

10. The <**AssertionIDRequestService>** element within **<AttributeAuthorityDescriptor>** MUST NOT be present.

11. One or two **<AttributeProfile>** elements MUST be present within **<AttributeAuthorityDescriptor>** to indicate that the AP supports encrypted or clear-text subject identifiers, or both, as follows:

    a. If the AP supports clear-text subject identifiers, then one of the **<AttributeProfile>** elements MUST have a value of "**urn:idmanagement.gov:icam:bae:v2:SAML:2.0:profiles:query:attribute:nameid-cleartext**".

    b. If the AP supports encrypted subject identifiers, then one of the **<AttributeProfile>** elements MUST have a value of "**urn:idmanagement.gov:icam:bae:v2:SAML:2.0:profiles:query:attribute:nameid-encrypted**".

5.2.6 <md:RoleDescriptor> Element Requirements

The following requirements apply to **<RoleDescriptor>** elements that appear in the NIEF SAML Cryptographic Trust Fabric document. Each **<RoleDescriptor>** element

provides metadata for a specific NIEF Web Services endpoint. These requirements supplement the requirements described in the trust fabric extension schema for NIEF Web Services[4] and in [SAML2 Meta AR Ext].

1. The **xsi:type** attribute within **<md:RoleDescriptor>** MUST be present, and its value MUST be specified in accordance with the following subsections.

   a. If the <RoleDescriptor> element describes a Web Service Consumer (WSC), then the **xsi:type** value MUST be "**gfipmws:GFIPMWebServiceConsumerType**".

   b. If the <RoleDescriptor> element describes a Web Service Provider (WSP), then the **xsi:type** value MUST be "**gfipmws:GFIPMWebServiceProviderType**".

   c. If the <RoleDescriptor> element describes an Assertion Delegate Service (ADS), then the **xsi:type** value MUST be "**gfipmws:GFIPMAssertionDelegateServiceType**".

   d. If the <RoleDescriptor> element describes an Attribute Consumer (AC), then the **xsi:type** value MUST be "**mdext:AttributeRequesterDescriptorType**".

2. The **protocolSupportEnumeration** attribute within **<md:RoleDescriptor>** MUST be present, and its value MUST consist of a list of one or more supported NIEF Web Services service interaction profiles (SIPs), delimited by spaces (" ") specified using the following URIs.[5,6]

   - **http://gfipm.net/standards/webservices/1.0/consumer-provider-sip.html**
   - **http://gfipm.net/standards/webservices/1.0/user-consumer-provider-sip.html**
   - **http://gfipm.net/standards/webservices/1.0/trusted-identity-broker-sip.html**
   - **http://gfipm.net/standards/webservices/1.0/saml-assertion-delegate-service-sip.html**

[4] See http://gfipm.net/standards/metadata/2.1/webservices.
[5] See [NIEF S2S] for information about each NIEF Web Services service interaction profile (SIP), including motivating use cases and normative language.
[6] The list of SIPs below includes only those SIPs for which normative language has been defined as of version 1.0 of [NIEF S2S]. Subsequent versions of [NIEF S2S] may contain normative language for additional SIPs, and when those SIPs are available for operational use, this document will be updated to indicate the appropriate URIs to use for them.

- **http://gfipm.net/standards/webservices/1.1/attribute-provider-sip.html**

3. Each **<RoleDescriptor>** element MAY contain one **<Extensions>** element, and the **<Extensions>** element MAY contain one **<mdattr:EntityAttributes>** element as defined by [SAML2 Entity Attr]. The **<mdattr:EntityAttributes>** element MAY contain zero or more SAML **<Attribute>** elements, in accordance with [SAML2 Entity Attr].

4. At least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**signing**" MUST be present within **<RoleDescriptor>**.

5. If the endpoint supports XML encryption, then at least one **<KeyDescriptor>** element containing a **use** attribute with a value of "**encryption**" MUST be present within **<RoleDescriptor>**.

6. Each **<KeyDescriptor>** element MUST include a **<ds:KeyInfo>** element containing exactly one **<ds:X509Data>** element, and the **<ds:X509Data>** element MUST contain exactly one **<ds:X509Certificate>** element. Other sub-elements of the **<KeyDescriptor>** element are permitted, but they MUST represent the same key.

7. If the **<RoleDescriptor>** element has a type of **mdext:AttributeRequesterDescriptorType**, then the additional requirements in the following subsections apply.

   a. The **WantAssertionsSigned** attribute MUST be present and its value MUST be "true".

   b. The **<RoleDescriptor>** element SHOULD, for descriptive and discovery purposes, contain at least one **<AttributeConsumingService>** element that contains a series of **<RequestedAttribute>** elements that denotes the set of attributes that the AC may request from an AP.

5.3 NIEF REST Cryptographic Trust Fabric Document

The NIEF REST Cryptographic Trust Fabric document is a JavaScript Object Notation (JSON) Web Token (JWT) document as defined in [JWT] and is also based on the JSON Resource Descriptor (JRD) format defined in Section 4.4 of [WebFinger]. It also uses several extensions to these JWT and JRD formats. The claims set of the NIEF REST

Cryptographic Trust Fabric JWT MUST conform to Sections 5.3.1 through 5.3.8. The JWT MUST be encoded and digitally signed in accordance with Section 5.3.9.

Each entry in the NIEF REST Cryptographic Trust Fabric document specifies an entity descriptor for a NIEF REST software entity that performs one or more NIEF REST roles. These roles are:

- OpenID Provider
- OpenID Connect (OIDC) Relying Party (RP)
- REST Assertion Delegate Service (ADS)
- REST Attribute Provider (AP)
- REST Authorization Service (AS)
- REST Service Consumer (RSC)
- REST Service Provider (RSP)

5.3.1 NIEF REST Cryptographic Trust Fabric JWT Claims Set

The claims set portion of the NIEF REST Cryptographic Trust Fabric JWT contains the trust fabric entries and is comprised of the following members. For each member, a flag is displayed within square brackets denoting whether the member is required to exist.

1. **`iss`** [REQUIRED] – This claim identifies the issuer of NIEF REST Cryptographic Trust Fabric document. It MUST conform to Section 4.1.1 of [JWT] and MUST denote the NIEF Center.

2. **`sub`** [REQUIRED] – This is the subject claim. Its semantics are as specified in Section 4.1.2 of [JWT] and its value MUST be the string "NIEF REST Cryptographic Trust Fabric".

3. **`exp`** [REQUIRED] – This claim specifies the expiration time of the Trust Fabric document and MUST conform to Section 4.1.4 of [JWT]. The JWT claims set also contains a set of entity descriptors and each descriptor contains an expiration time. The value of this claim MUST NOT be earlier than any expiration time of any descriptor. The value of this claim SHOULD match the expiration time of the descriptor that has the latest expiration time.

4. **`iat`** [REQUIRED] – This claim specifies the time at which the Trust Fabric document was issued and MUST conform to Section 4.1.6 of [JWT].

5. **`jti`** [REQUIRED] – This claim specifies a unique identifier for a particular instance of the Trust Fabric document and MUST conform to Section 4.1.7 of [JWT].[7]

[7] No two instances of the NIEF REST Cryptographic Trust Fabric document should have the same jti value.

6. **`entities`** [REQUIRED] – **`entities`** is a private claim name in accordance with Section 4.3 of [JWT][8]. The value of this claim MUST be an array with each element of the array being a JRD object in accordance with Section 4.4 of [WebFinger] and with Section 5.3.2 of this document. Each JRD specifies a Trust Fabric entity descriptor for a NIEF REST software entity.

5.3.2 General JRD Requirements

Each JRD in the Trust Fabric MUST conform to the following requirements.

1. Each JRD MUST contain the following members. For each member, a flag is displayed within square brackets denoting whether the member is required to exist.

    a. **`subject`** [REQUIRED] – The value of this member MUST be a URI that is controlled by the organization that owns this entity.

        i. If this descriptor describes an OpenID Provider (including OIDC IDPs, REST ADSes, and REST APs), then this value MUST be the entity's issuer value as used in [OIDC Core], or the relevant NIEF REST SIPs. In addition, this value MUST be a URL that uses the "https" scheme with no query or fragment component.

        ii. If this descriptor describes an OAuth Authorization Server (including REST ASes), then this value MUST be a URL that uses the "https" scheme with no query or fragment component.

        iii. If this descriptor describes an OIDC RP, OAuth Client, or RSC, then this value MUST be the entity's client identifier as used in [OIDC Core], [OAuth Core], or the relevant NIEF REST SIPs.

        iv. If this descriptor describes an RSP, then this value MUST be a base URI of all resources hosted at the RSP.

        v. A **`subject`** value of any descriptor MUST NOT be a base URI for the **`subject`** value of another descriptor.

    b. **`exp`** [REQUIRED] – This member specifies the time after which the information in the descriptor should not be trusted. The value of this member MUST conform to the requirements for the **`exp`** JWT claim as specified in Section 4.1.4 of [JWT].

    c. **`org`** [REQUIRED] – This member contains information about the organization that controls the entity. The value MUST be an object with the

[8] Private claim names are names that are not registered in accordance with [JWT] and are not collision-resistent names.

following members.; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

i. **`name`** [REQUIRED] – The value of this member MUST be a string that is the name of the organization.

ii. **`url`** [REQUIRED] – The value of this member MUST be a string that is a URL at which more information about the organization can be obtained.

iii. **`desc`** [REQUIRED] – The value of this member MUST be a string that is a brief description of the organization.

d. **`pocs`** [REQUIRED] – This member contains information about a set of points of contact for the entity. The value MUST be an array that contains at least one entry and in which each entry is an object that has the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

i. **`name`** [REQUIRED] – The value of this member MUST be a string that denotes the name of the point of contact.

ii. **`org_name`** [OPTIONAL] – The value of this member MUST be a string that denotes the name of the contact's organization.

iii. **`email`** [REQUIRED] – The value of this member MUST be a string that denotes the email address of the contact.

iv. **`tel`** [OPTIONAL] – The value of this member MUST be a string that denotes the telephone number of the contact.

e. **`links`** [REQUIRED] – The semantics of this member are as defined in Section 4.4.4 of [WebFinger], in addition to the requirements in the following sub-items.

i. The value of this member MUST be an array in which each entry is an object that is a "link relation object" where each link relation object contains metadata about a particular type of NIEF REST endpoint.

ii. The array MUST contain at least one link relation object.

f. **`jwks`** [CONDITIONAL] – The value of this member MUST be the entity's JSON Web Key (JWK) Set (see [JWK]) document. The JWK Set MUST contain at least one signing key and MAY contain one or more encryption keys. When both signing and encryption keys are made available, a **`use`**

(Key Use) parameter value is REQUIRED for all keys in the referenced JWK Set to indicate each key's intended usage. The same key MUST NOT be used for both signatures and encryption. The JWK **`x5c`** parameter MAY be used to provide X.509 representations of keys provided. When the **`x5c`** parameter is used, the bare key values MUST still be present and MUST match those in the certificate.

This member MUST exist unless one of the following conditions hold.

i. The entity is only an RSC and the only protocol supported by the entity is the REST SSO-Consumer-Authorizer SIP.[9]

ii. The entity is only an OIDC RP and the only protocol supported by the entity is the OIDC SSO SIP.[10]

g. **`attrs`** [OPTIONAL] – The value of this member MUST be an object in which each member represents an attribute assigned to the entity. Such attributes MAY come from the NIEF Attribute Registry (see [NIEF Attr]). Attributes from the NIEF Attribute Registry MUST be expressed in accordance with the NIEF Attribute Encoding Rules (see [NIEF Attr Enc]).

h. The JRD MAY contain other members.

2. If the JRD describes an OpenID Provider, including OIDC IDPs, REST ADSes, and REST APs, then the JRD MUST also conform to Section 5.3.3.

3. If the JRD describes an OAuth Authorization Server, including REST ASes, then the JRD MUST also conform to Section 5.3.4.

4. If the JRD describes an OIDC RP, including NIEF REST endpoints that act as OIDC RPs, then the JRD MUST also conform to Section 5.3.5.

5. If the JRD describes an RSC (including an RSC that acts as an OAuth Client), then the JRD MUST also conform to Section 5.3.6.

6. If the JRD describes an RSP, then the JRD MUST also conform to Section 5.3.8.

5.3.3 OpenID Provider Descriptor Requirements

This section applies to OpenID Providers, including OIDC IDPs, REST ADSes, and REST APs. A JRD that describes an OpenID Provider MUST conform to the following requirements.

[9] In this case, the RSC acts as an OAuth Client. If the “jwks” member is not present in this case, then trust is based in the “redirect_uris” member in the link relation object for the Client.
[10] In this case, if the “jwks” member is not present, then trust is based in the “redirect_uris” member in the link relation object for the RP.

1. The **`links`** array of a JRD that describes an OpenID Provider MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

    a. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "http://openid.net/specs/connect/1.0/issuer".

    b. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be a string that matches the **`subject`** member of the parent object. The value of this member MUST be the issuer value of any assertion asserted by the OpenID Provider.

    c. The link relation object MAY include other members that are defined in Section 3 of [OIDC Disc].

    d. If the **`issuer`** member is present, then its value MUST match the value of the link relation object's **`href`** member and the JRD's **`subject`** member.

    e. If the **`jwks_uri`** member is present, then the value of this member MUST be a URL that points to a JWK Set that is the same as the JWK Set in the parent JRD.

    f. If the **`token_endpoint_auth_methods_supported`** member is present, then its array MAY include the client authentication methods declared in Section 5.3.9.

    g. If the **`response_types_supported`** member is present, then its array MAY include response types declared in Section 5.3.12.

    h. If the **`grant_types_supported`** member is present, then its array MAY include grant types declared in Section 5.3.13.

5.3.4 OAuth Authorization Server Descriptor Requirements

A JRD that describes an OAuth Authorization Server MUST conform to the following requirements.

1. The **`links`** array of the JRD MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

a. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "https://nief.org/specs/rest/1.0/rest-as".

b. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be a string that matches the **`subject`** member of the parent object.

c. **`authorization_endpoint`** [OPTIONAL] – The value of this member MUST be a string that is the URL of the Server's OAuth Authorization Endpoint

d. **`token_endpoint`** [OPTIONAL] – The value of this member MUST be a string that is the URL of the Server's OAuth Token Endpoint.

e. **`registration_endpoint`** [OPTIONAL] – The value of this member MUST be a string that is the URL of the Server's OAuth Dynamic Client Registration Endpoint.

f. **`scopes_supported`** [OPTIONAL] – The value of this member MUST be an array containing a list of OAuth scope values that the Server supports. This list of scope values MAY exclude some supported values.

g. **`grant_types_supported`** [OPTIONAL] – The value of this member MUST be an array containing a list of the OAuth grant type parameter values that the Server supports. Valid values are defined in Section 5.3.11.

h. **`response_types_supported`** [OPTIONAL] – The value of this member MUST be an array containing a list of the OAuth response type parameter values that the Server supports. Valid values are defined in Section 5.3.10.

i. **`token_endpoint_auth_methods_supported`** [OPTIONAL] – The value of this member MUST be an array containing a list of client authentication methods supported by the Server's Token Endpoint. Valid array values include the client authentication methods declared in Section 5.3.9.

j. **`token_endpoint_auth_signing_alg_values_supported`** [OPTIONAL] – The value of this member MUST be a JSON array containing a list of the JWS signing algorithms (alg values) supported by the Token Endpoint for the signature on the JWT [JWT] used to authenticate the Client at the Token Endpoint for the "private_key_jwt" and "client_secret_jwt" authentication methods. The value "none" MUST NOT be used.

5.3.5 OpenID Connect (OIDC) Relying Party (RP) Descriptor Requirements

This section applies to OIDC RPs, and RSCs that act as REST ACs or OIDC RPs. A JRD that describes an OIDC RP MUST conform to the following requirements.

1. The **`links`** array of a JRD that describes an OIDC RP MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

    a. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "https://nief.org/specs/rest/1.0/oidc-rp".

    b. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger], in addition to the following requirements. The value of this member MUST be a string that matches the **`subject`** member of the parent object.

    c. The link relation object MAY include other members defined in Section 2 of [OIDC DCR].

    d. If the **`jwks`** member of the parent JRD is not present, then the **`redirect_uris`** member, as defined in Section 2 of [OIDC DCR] MUST be present here.

    e. If the **`token_endpoint_auth_method`** member is present, then the value of this member MAY include values specified in Section 5.3.9.

    f. If the **`response_types`** member is present, then the **`response_types`** array MAY include values specified in Section 5.3.12.

    g. If the **`grant_types`** member is present, then the **`grant_types`** array MAY include values specified in Section 5.3.13.

    h. If the **`jwks_uri`** member is present, then the value of this member MUST be a URL that points to a JWK Set that is the same as the JWK Set in the parent JRD.

    i. If the **`jwks`** member is present, then the value of this member MUST be a JWK Set that is the same as the JWK Set in the parent JRD.

5.3.6 OAuth Client Descriptor Requirements

This section applies to RSCs that act as OAuth Clients. A JRD that describes an OAuth Client MUST conform to the following requirements.

1. The **`links`** array of a JRD that describes an OAuth Client MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

    a. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "https://nief.org/specs/rest/1.0/oauth-client".

    b. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger], in addition to the following requirements. The value of this member MUST be a string that matches the **`subject`** member of the parent object.

    c. The link relation object MAY include other members defined in Section 2 of [OAuth DCR].

    d. If the **`jwks`** member of the parent JRD is not present, then the **`redirect_uris`** member, as defined in Section 2 of [OAuth DCR] MUST be present.

    e. If the **`token_endpoint_auth_method`** member is present, then the value of this member MAY include values specified in Section 5.3.9.

    f. If the **`response_types`** member is present, then the **`response_types`** array MAY include values specified in Section 5.3.10.

    g. If the **`grant_types`** member is present, then the **`grant_types`** array MAY include values specified in Section 5.3.11.

    h. If the **`jwks_uri`** member is present, then the value of this member MUST be a URL that points to a JWK Set that is the same as the JWK Set in the parent JRD.

    i. If the **`jwks`** member is present, then the value of this member MUST be a JWK Set that is the same as the JWK Set in the parent JRD.

5.3.7 REST Service Consumer Descriptor Requirements

A JRD that describes a REST Service Consumer (RSC) MUST conform to the following requirements.

1. The **`links`** array of a JRD that describes a REST Service Consumer (RSC) MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

2. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "https://nief.org/specs/rest/1.0/rsc".

3. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger], in addition to the following requirements. The value of this member MUST be a string that matches the **`subject`** member of the parent object.

5.3.8 REST Service Provider Descriptor Requirements

A JRD that describes a REST Service Provider (RSP) MUST conform to the following requirements.

1. The **`links`** array MUST include a link relation object that contains the following members; for each member, a flag is displayed within square brackets denoting whether the member is required to exist.

    a. **`rel`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger]. The value of this member MUST be "https://nief.org/specs/rest/1.0/rsp".

    b. **`href`** [REQUIRED] – This member has semantics as defined in Section 4.4.4 of [WebFinger], in addition to the following requirements. The value of this member MUST be a string that matches the **`subject`** member of the parent object.

2. If the RSP accepts OAuth access tokens from a REST AS whose issuer identifier is equivalent to the Resource Server's base URL, then this JRD SHOULD also conform to Section 5.3.4 with respect to the REST AS.

3. If the RSP accepts OAuth access tokens from REST ASes whose issuer identifiers are different than the resource server's base URL, then for each AS, the following requirements SHOULD be met for discovery purposes.

    a. There SHOULD be a link relation object in the **`links`** array that conforms to the following requirements.

        i. The **`rel`** member MUST exist and MUST have a value of "https://nief.org/specs/rest/1.0/as"

        ii. The **`href`** member MUST exist and its value MUST be the issuer value of the REST AS.

    b. There SHOULD be an OAuth Authoriztion Server entry in the Trust Fabric for the AS.

5.3.9 Supported OAuth Client Authentication Methods

This section enumerates the supported methods for an OAuth Token Endpoint to authenticate an OAuth Client.

1. "client_secret_post", "client_secret_basic", "client_secret_jwt", and "private_key_jwt", in accordance with Section 9 of [OIDC Core].

2. "client_secret_saml" denotes SAML bearer token authentication in accordance with [OAuth SAML2] where the SAML assertion is protected via a message authentication code.

3. "private_key_saml" denotes SAML bearer token authentication in accordance with [OAuth SAML2] where the SAML assertion is protected via a digital signature.

4. "tls_cca" denotes TLS with client certificate authentication.

5.3.10 Supported OAuth Response Types

This section enumerates the OAuth response types that are supported in NIEF REST profiles of OAuth.

1. "code" in accordance with Section 2 of [OAuth DCR].

2. "token" in accordance with Section 2 of [OAuth DCR].

5.3.11 Supported OAuth Grant Types

This section enumerates the OAuth grant types that are supported in NIEF REST profiles of OAuth.

1. "implicit" in accordance with Section 2 of [OAuth DCR].

2. "authorization_code" in accordance with Section 2 of [OAuth DCR].

3. "client_credentials" in accordance with Section 2 of [OAuth DCR].

4. "refresh_token" in accordance with Section 2 of [OAuth DCR].

5. "urn:ietf:params:oauth:grant-type:jwt-bearer" in accordance with [OAuth JWT].

6. "urn:ietf:params:oauth:grant-type:saml2-bearer" in accordance with [OAuth SAML2].

5.3.12Supported OpenID Connect Response Types

This section enumerates the OAuth response types that are supported in NIEF REST profiles of OIDC.

1. “code” in accordance with Section 2 of [OIDC DCR].
2. “token” in accordance with Section 2 of [OIDC DCR].
3. “id_token” in accordance with Section 2 of [OIDC DCR].
4. “delegated_token” in accordance with the NIEF REST ADS SIP defined in Section 6.8 of [NIEF REST].

5.3.13Supported OpenID Connect Grant Types

This section enumerates the OAuth grant types that are supported in NIEF REST profiles of OAuth.

1. “implicit” in accordance with Section 2 of [OAuth DCR].
2. “authorization_code” in accordance with Section 2 of [OAuth DCR].
3. “refresh_token” in accordance with Section 2 of [OAuth DCR].
4. “urn:ietf:params:oauth:grant-type:jwt-bearer” in accordance with [OAuth JWT].
5. “urn:ietf:params:oauth:grant-type:saml2-bearer” in accordance with [OAuth SAML2].

5.3.14 NIEF REST Cryptographic Trust Fabric Encoding Requirements

The NIEF REST Cryptographic Trust Fabric is a JWT that is digitally signed via JWS. In addition, the NIEF Center MAY distribute individual JRD entity descriptors as standalone documents outside of the NIEF REST Cryptographic Trust Fabric. A standalone JRD entity descriptor MUST be the claims set portion of a JWT that is digitally signed via JWS.

For both the NIEF REST Cryptographic Trust Fabric and individual JRD documents, prior to executing the signing operation, all leading whitespace, all trailing whitespace, and all whitespace in between the JSON tokens of the JWT claims set and JOSE header MUST be removed, where token and whitespace are as defined in [JSON].

5.4 Duplicate Entity Identifiers in the NIEF Cryptographic Trust Fabric

A single entity identifier for a software entity MAY appear in both the SAML and REST NIEF Cryptographic Trust Fabric documents. However, An entity identifier MUST NOT appear more than once within the NIEF SAML Cryptographic Trust Fabric document and

MUST NOT appear more than once with the NIEF REST Cryptographic Trust Fabric document.

5.5 Definition of Base URI

When used in this document, the term "base URI" is defined as follows. A URI (the first URI) is a base URI of a second URI if the following conditions hold.

1. The scheme part of both URIs are equivalent.

2. The authority part of both URIs are empty, or the authority part of both URIs are equivalent.

3. The path component of the second URI includes the path component of the first URI. This condition holds when the path component of both URIs are equivalent.

4. The query and fragment parts are not considered in this comparison.

5.5.1 Examples (Nonnormative)

1. "https://nief.org/trust/" is a base URI of "https://nief.org/trust/example/one/".

2. "https://nief.org/trust" is NOT a base URI of "https://sub.nief.org/trust/example/one/", since the authority component of both URIs are not equivalent.

5.6 Trust Fabric Lifecycle Management Procedures

This section describes policies and procedures used to manage the NIEF Cryptographic Trust Fabric ("Trust Fabric"). It includes details about how the Trust Fabric is created and distributed, as well as the conditions under which the Trust Fabric is updated.

5.6.1 Providing Federated System Entity Metadata to the NIEF Center

The NIEF Cryptographic Trust Fabric documents are produced by the NIEF Center and contain metadata about each system endpoint within NIEF. It is the responsibility of the NIEF Center to keep the NIEF Cryptographic Trust Fabric documents up-to-date with the correct information for all NIEF system endpoints.

To enable the NIEF Center to keep the NIEF Cryptographic Trust Fabric documents up-to-date, each IDPO, APO, SPO, and SCO in NIEF MUST provide the necessary metadata to the NIEF Center prior to its initial participation in NIEF, and on an ongoing basis thereafter any time that the metadata changes.

The NIEF Center SHALL notify each NIEF participating organization about the specific metadata that the member must keep up-to-date. This metadata typically differs based

on the type of member organization (IDPO, APO, SPO, or SCO) and the specific systems and system endpoints implemented by the member organization.

5.6.2 NIEF Trust Fabric Creation Procedure (Nonnormative)

Upon the occurrence of a triggering condition for the update of a Trust Fabric document (see
Section 5.6.4), the NIEF Center regenerates the document. The process of generating a new Trust Fabric document consists of two basic operations: editing the document to reflect the desired policy change (e.g., new IDP added to NIEF) and digitally signing the new document with the NIEF Center's Trust Fabric signing key. The following steps describe the basic editing process for a NIEF Cryptographic Trust Fabric document.

1. Starting with the most recent NIEF Cryptographic Trust Fabric document, edit the document as needed to incorporate the necessary changes.

2. Copy the edited NIEF Cryptographic Trust Fabric document to a USB flash token.

3. Connect the flash token containing the unsigned NIEF Cryptographic Trust Fabric document to the physical machine on which the signing operation will be performed. Also connect the USB flash token containing the NIEF Center's Trust Fabric signing key to the machine.[11]

4. Perform the cryptographic signing operation on the NIEF Cryptographic Trust Fabric document using the NIEF Center's Trust Fabric signing key. At no point during this operation is the NIEF Center's Trust Fabric signing key copied from the flash token onto any other storage device. Also, at no point during this operation is the physical machine connected to a network.

5. Copy the signed NIEF Cryptographic Trust Fabric document onto the USB flash token that contains the unsigned NIEF Cryptographic Trust Fabric document.

5.6.3 NIEF Trust Fabric Distribution Procedure (Nonnormative)

Upon the occurrence of a triggering condition for the update of a Trust Fabric document (see Section 5.6.4), and after the generation and signing of a new Trust Fabric document (see Section 5.6.2), the NIEF Center distributes an updated version of the document to all NIEF participants. The following steps describe how the basic NIEF Cryptogrphic Trust Fabric document distribution process works.

[11] See [NIEF CP] for a more detail about how NIEF's Trust Fabric signing key is managed.

1. Publish the new NIEF Cryptographic Trust Fabric document at the appropriate well-known URL.[12]

2. Notify all NIEF participants of the new NIEF Cryptographic Trust Fabric document via the technical contact points they have provided.

Note that while the integrity of each NIEF Cryptographic Trust Fabric document is paramount to the security of NIEF, each Trust Fabric document need not necessarily be kept confidential. Therefore, it is permissible for the Trust Fabric URLs to be publicly accessible, and encryption of the Trust Fabric documents is not necessary.

5.6.4 Triggering Conditions for NIEF Trust Fabric Updates

The NIEF Center MUST regenerate and redistribute one or both of the NIEF Cryptographic Trust Fabric documents upon the occurrence of any of the following events.

1. A new system entity begins participating in NIEF.

2. An existing system entity withdraws from participation in NIEF.

3. An existing NIEF system entity undergoes a configuration change that affects its entry in the trust fabric (e.g., certificate expiration, migration to a new server, key compromise on a server, etc.).

4. The NIEF Center's Trust Fabric signing key certificate expires.

5. It is suspected that the NIEF Center's Trust Fabric signing key has been compromised.

6. One or both of the current Trust Fabric documents have expired or are due to expire in the very near future.

Note that (1) and (2) are usually (but not always) caused when an organization begins participating in NIEF or withdraws from NIEF.

5.6.5 Import and Consumption of Trust Fabric by NIEF Members

A NIEF Metadata-Consuming System is any system that implements one or more NIEF-facing endpoints and relies on the NIEF Cryptographic Trust Fabric as a basis for its cryptographic trust decisions.

[12] See Appendix B for the current URLs at which the NIEF Cryptographic Trust Fabric documents are published.

1. A NIEF Metadata-Consuming System MUST support at least one of the following mechanisms for automated import of NIEF Cryptographic Trust Fabric documents:

    a. Automated import from a remote resource at a fixed location accessible via HTTP 1.1 over TLS 1.1 or higher; or

    b. Automated import from a local file obtained out-of-band.

2. At trust fabric consumption time, a NIEF Metadata-Consuming System MUST perform signature verification at the root level of the NIEF Cryptographic Trust Fabric document of interest, and MUST NOT import the contents of the document unless the document was signed by the NIEF Center's trust fabric signing key.

3. Prior to trusting metadata in a NIEF Cryptographic Trust Fabric document, a NIEF Metadata-Consuming System MUST honor the expiration times of the document and of every entity descriptor contained within, as specified by the **`validUntil`** attribute of all **`<EntitiesDescriptor>`** and **`<EntityDescriptor>`** elements in the NIEF SAML Cryptographic Trust Fabric document, and by the **`exp`** JSON member of the root and JRD objects in the NIEF REST Cryptographic Trust Fabric document, and attempt to refresh the document before it expires. If the expiration time has passed for a Trust Fabric document or particular entity descriptor, then the system MUST discontinue trusting the affected entities until it obtains a new document with updated expiration times. In addition, prior to trusting metadata in the NIEF SAML Cryptographic Trust Fabric document, a NIEF Metadata-Consuming System MUST honor the cache duration period of the document and of every entity descriptor contained within, as specified by the **`cacheDuration`** attribute of all **`<EntitiesDescriptor>`** and **`<EntityDescriptor>`** elements, and attempt to refresh the document before the cache duration period has passed since the last refresh. If the document cannot be refreshed before its **`cacheDuration`** expires, the system MUST make a risk-based determination about whether to continue transacting with the affected entities.

4. A NIEF Metadata-Consuming System SHOULD be capable of processing a NIEF SAML Cryptographic Trust Fabric document that contains one or more **`<EntitiesDescriptor>`** elements nested within another **`<EntitiesDescriptor>`** element.

5.7 Standard NIEF Trust and Security Considerations

This section provides basic normative rules regarding the use of cryptography for messages sent within all NIEF communication profiles. Message senders and recipients MUST obey these rules at all times, unless directed otherwise by a specific communication profile.

5.7.1 Digital Signature Creation and Processing

A message sender MUST sign all messages, or the appropriate parts thereof according to the rules of the applicable NIEF communication profile, using the sender's digital signature certificate that either appears in the relevant NIEF Cryptographic Trust Fabric entity descriptor, or has been registered with the recipient via boostrapping the relevant NIEF Cryptographic Trust Fabric entity descriptor. The digital signature allows the recipient of the message to authenticate the sender and confirm that the message has not been altered since the time at which the signature was applied.

1. The recipient MUST authenticate the sender and verify the signature upon receipt of the message.

2. The recipient MUST verify that the sender of the message is included in the relevant NIEF Cryptographic Trust Fabric document.

3. If inclusion in the relevant NIEF Cryptographic Trust Fabric document cannot be determined for the message sender, then the message recipient MUST reject the message.

5.7.2 Message Encryption

Encryption ensures that only the intended recipient can decipher the message and gain access to confidential information in it.

1. For all NIEF communication profiles, all confidential information in a message MUST be encrypted according to the rules of the applicable communication profile.

2. Unless otherwise stipulated by the applicable NIEF communication profile, encryption MUST use the public key of one of the intended recipient's encryption certificates as it appears in the recipient's entity descriptor in the relevant NIEF Cryptographic Trust Fabric document.

5.7.3 Minimum Requirements for Cryptographic Algorithms and Modules

For all NIEF communications, the following cryptographic algorithm and module requirements are in force.

1. For symmetric key encryption functions, all communications MUST use AES with keys of 128 bits keys or longer, or a stronger [FIPS 140-2] approved algorithm.

2. For hashing functions, all communications MUST use SHA-256, SHA-384, or SHA-512, or a stronger [FIPS 140-2] approved algorithm.

3. For public-key encryption and signing functions, all communications MUST use RSA or a stronger [FIPS 140-2] approved algorithm.

4. All NIEF-facing systems MUST use [FIPS 140-2] validated cryptographic modules for all encryption and digital signature functions.

5.8 Other NIEF Reference Documents (Nonnormative)

This document does not represent the complete set of requirements for participation in NIEF. Other documents may apply, including business and policy documents (e.g., [NIEF Bylaws] and [NIEF OPP]), laws and regulations (e.g., [NIST SP 800-63-2]), and applicable technology standards (e.g., XML standards).

Appendix A: Extension Schema for <md:RoleDescriptor>

The diagram below contains the SAML Metadata extension schema for the **<md:RoleDescriptor>** element, which accommodates the inclusion of NIEF Web Services endpoints within a NIEF Cryptographic Trust Fabric document.

```
<?xml version="1.0" encoding="US-ASCII"?>
<xs:schema targetNamespace="http://gfipm.net/standards/metadata/2.1/webservices"
           xmlns:xs="http://www.w3.org/2001/XMLSchema"
           xmlns:gfipmws="http://gfipm.net/standards/metadata/2.1/webservices"
           xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
           xmlns:wsa="http://www.w3.org/2005/08/addressing"
           elementFormDefault="unqualified"
           attributeFormDefault="unqualified"
           version="1.0">

    <xs:import namespace="urn:oasis:names:tc:SAML:2.0:metadata"
               schemaLocation="saml-schema-metadata-2.0.xsd" />
    <xs:import namespace="http://www.w3.org/2005/08/addressing"
               schemaLocation="ws-addr.xsd" />

    <!-- GFIPM Web Service Provider -->
    <xs:complexType name="GFIPMWebServiceProviderType">
        <xs:complexContent>
            <xs:extension base="gfipmws:WebServiceDescriptorType">
                <xs:sequence>
                    <xs:element ref="gfipmws:WebService" minOccurs="1" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

    <!-- GFIPM Security Token Service which currently works as an Assertion Delegate Service -->
    <xs:complexType name="GFIPMSecurityTokenServiceType">
        <xs:complexContent>
            <xs:extension base="gfipmws:WebServiceDescriptorType">
                <xs:sequence>
                    <xs:element ref="gfipmws:SecurityTokenService" minOccurs="1"
maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

    <!-- GFIPM Web Service Consumer -->
    <xs:complexType name="GFIPMWebServiceConsumerType">
        <xs:complexContent>
            <xs:extension base="gfipmws:WebServiceDescriptorType">
                <xs:sequence>
                    <xs:element ref="gfipmws:WebService" minOccurs="0" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

    <!-- Based on Section 3.1.2.1 from the WS-Federation Schemas -->
    <xs:complexType name="WebServiceDescriptorType" abstract="true">
        <xs:complexContent>
            <xs:extension base="md:RoleDescriptorType"/>
        </xs:complexContent>
    </xs:complexType>

    <xs:complexType name="WebServiceType">
        <xs:sequence>
            <xs:element ref="gfipmws:ServiceDisplayName" minOccurs="0" maxOccurs="1"/>
```

```
            <xs:element ref="gfipmws:ServiceDescription" minOccurs="0" maxOccurs="1"/>
            <xs:element ref="gfipmws:WebServiceEndpoint" minOccurs="1" maxOccurs="1"/>
            <xs:element ref="gfipmws:MetadataExchangeEndpoint" minOccurs="0" maxOccurs="1"/>
            <xs:element ref="gfipmws:WSDLEndpoint" minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
        <xs:attribute name="endpointID" type="md:entityIDType" use="required"/>
    </xs:complexType>

    <xs:element name="WebService"  type="gfipmws:WebServiceType" />
    <xs:element name="SecurityTokenService" type="gfipmws:WebServiceType" />

    <xs:element name="WebServiceEndpoint" type="wsa:EndpointReferenceType"/>
    <xs:element name="MetadataExchangeEndpoint" type="wsa:EndpointReferenceType"/>
    <xs:element name="WSDLEndpoint" type="wsa:EndpointReferenceType"/>

    <xs:element name="ServiceDisplayName" type="xs:string"/>
    <xs:element name="ServiceDescription" type="xs:string"/>

</xs:schema>
```

Figure 2: Extension Schema for `<md:RoleDescriptor>`

Appendix B: NIEF Cryptographic Trust Fabric URL

The most recent versions of all available NIEF Cryptographic Trust Fabric documents are available at the following URL.

https://nief.org/trust-fabric/