Retrieving Information Card Metadata via HTTPS GET

October 2008

Author

Michael B. Jones, Microsoft Corporation

Copyright Notice

Abstract

The Identity Metasystem allows users to manage their Digital Identities from various Identity Providers and employ them in different contexts where they are accepted to access online services. In the Identity Metasystem, identities are represented to users as "Information Cards".

This document supplements the information provided in three other Identity Selector Interoperability Profile references: the "Identity Selector Interoperability Profile V1.5" [ISIP], which provides the normative schema definitions and behaviors for Information Cards and the interoperable Identity Selectors that use them, "An Implementer's Guide to the Identity Selector Interoperability Profile V1.5" [ISIP Guide], which provides a non-normative description of the overall Information Card Model, and "A Guide to Using the Identity Selector Interoperability Profile V1.5 within Web Applications and Browsers" [ISIP Web Guide], which describes how Information Cards can be used within applications hosted on web sites and accessed through web browsers.

This document describes an additional mechanism that can be employed by software implementing the Information Card Model to communicate Information Card Metadata: HTTPS GET.

STATUS

The information presented in this document is informative; the normative definitions can be found in [ISIP]. This document supplements the information presented in [ISIP Guide].

Table of Contents

1. Introduction

In this document augments the information provided by [ISIP], [ISIP Guide], and [ISIP Web Guide] on how Security Token Services (STSs) implementing the Information Card Model can advertise metadata about their endpoints and how Identity Selectors can retrieve that metadata. In places where statements in this document conflict with statements in the [ISIP Guide], the statements in this document take precedence.

In order to support Information Cards, a Relying Party using Web-services based application will need to:

- Support the retrieval of its service metadata using either HTTPS GET, as described in this document, or the mechanism described in [WS-MetadataExchange], or both.

In order to support Information Cards, an Identity Provider will need to:

- Support the retrieval of its service metadata using either HTTPS GET, as described in this document, or the mechanism described in [WS-MetadataExchange], or both.

For brevity of the examples used for illustration in this document, the XML namespace prefixes used in this document are the same as those in Table 1 of [ISIP Guide].

2. Retrieving Identity Provider STS Metadata via HTTPS GET

Section 4.1.1.2 of the [ISIP] specification states that "An Identity Selector MAY retrieve the Security Policy it will use to communicate with the IP/STS from that metadata location using the mechanism specified in [WS-MetadataExchange]". This section describes another possible mechanism: retrieving the metadata document directly from the advertised metadata location using HTTPS GET.

Identity Providers MAY publish metadata for Information Cards as WSDL documents that can be retrieved by Identity Selectors via HTTPS GET operations on URLs using the HTTPS scheme. And endpoint's metadata URL is communicated to Identity Selectors in a token service `wsx:MetadataReference` element in an Information Card using exactly the same syntax as when MEX is employed to retrieve the metadata. No change occurs in the card.

The metadata documents published via HTTPS GET SHOULD contain the WSDL for the endpoint as the top-level element of the document without any SOAP or MEX elements enclosing it.

Identity Providers MAY publish endpoint metadata via both the HTTPS GET and MEX methods at the same metadata location. If they publish the metadata via multiple mechanisms, the metadata delivered via both mechanisms SHOULD be the same. Likewise,

Identity Selectors MAY attempt to retrieve metadata via multiple mechanisms, either in sequence or in parallel.

While the [ISIP Guide] stated that MEX must be used, this document supersedes those sections of the guide by introducing another possible retrieval mechanism. Therefore, the "must" statements about the use of MEX in the guide should instead be treated as "may" statements. Note that there is already precedent for allowing multiple mechanisms for retrieving endpoint metadata. [WS-Federation] Section 3.2 "Acquiring the Federation Metadata Document" describes how WS-Federation implementations should first attempt to retrieve metadata using HTTPS GET and then via MEX.

2.1. Expressing Token Service Metadata Endpoints

The following example (taken verbatim from the [ISIP Guide]) illustrates an Identity Provider with two endpoints for its IP/STS, one requiring Kerberos (higher priority) and the other requiring username/password (lower priority) as its authentication mechanism. Note that the metadata endpoint locations are specified using exactly the same syntax no matter which retrieval mechanism is employed by the Identity Selector and the Identity Provider.

Example:

```
<ic:TokenServiceList>
  <ic:TokenService>
    <wsa:EndpointReference>
      <wsa:Address>http://contoso.com/sts/kerb</wsa:Address>
      <wsa:Metadata>
        <wsx:Metadata>
          <wsx:MetadataSection
              Dialect="http://schemas.xmlsoap.org/ws/2004/09/mex">
            <wsx:MetadataReference>
              <wsa:Address>https://contoso.com/sts/kerb/mex</wsa:Address>
            </wsx:MetadataReference>
          </wsx:MetadataSection>
        </wsx:Metadata>
      </wsa:Metadata>
    </wsa:EndpointReference>
    <ic:UserCredential>
      <ic:KerberosV5Credential />
    </ic:UserCredential>
  </ic:TokenService>
  <ic:TokenService>
    <wsa:EndpointReference>
      <wsa:Address>http://contoso.com/sts/pwd</wsa:Address>
      <wsa:Metadata>
        <wsx:Metadata>
          <wsx:MetadataSection
              Dialect="http://schemas.xmlsoap.org/ws/2004/09/mex">
            <wsx:MetadataReference>
              <wsa:Address>https://contoso.com/sts/pwd/mex</wsa:Address>
            </wsx:MetadataReference>
          </wsx:MetadataSection>
        </wsx:Metadata>
      </wsa:Metadata>
    </wsa:EndpointReference>
    <ic:UserCredential>
      <ic:UsernamePasswordCredential>
        <ic:Username>Zoe</ic:Username>
      </ic:UsernamePasswordCredential>
```

```
    </ic:UserCredential>
  </ic:TokenService>
</ic:TokenServiceList>
```

3. Retrieving Relying Party STS Metadata via HTTPS GET

Like IP/STSs, RP/STSs also publish endpoint metadata. This metadata CAN be retrieved via HTTPS GET in the same manner that IP/STS metadata can be.

Like IP/STSs, no changes to the syntax used to specify metadata locations occur when RP/STS metadata is published by the Relying Party STS as a page retrievable using HTTPS GET. Relying Parties and Identity Providers MAY consequently support either or both retrieval methods for the same metadata addresses.

4. References

[ISIP]

A. Nanda and M. Jones, "Identity Selector Interoperability Profile V1.5", July 2008.

[ISIP Guide]

Microsoft Corporation and Ping Identity Corporation, "An Implementer's Guide to the Identity Selector Interoperability Profile V1.5", July 2008.

[ISIP Web Guide]

M. Jones, "A Guide to Using the Identity Selector Interoperability Profile V1.5 within Web Applications and Browsers", July 2008.

[WS-Federation]

M. Goodner et al., "Web Services Federation Language (WS-Federation) Version 1.2, Committee Draft 02", January 2009.

[WS-MetadataExchange]

"Web Services Metadata Exchange (WS-MetadataExchange), Version 1.1", August 2006.

Appendix A – Example IP/STS Metadata Retrieved via HTTPS GET

An example metadata document retrieved from https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/mex using HTTPS GET follows:

```
<?xml version="1.0" encoding="utf-8" ?>
<wsdl:definitions name="SecurityTokenService"
targetNamespace="http://schemas.microsoft.com/ws/2008/06/identity/securitytokenservice"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:tns="http://schemas.microsoft.com/ws/2008/06/identity/securitytokenservice"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:i0="http://tempuri.org/"
xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:wsa10="http://www.w3.org/2005/08/addressing"
xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex">
<wsp:Policy wsu:Id="transportBindingConfig_IWSTrustFeb2005Sync_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:TransportBinding xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy>
          <sp:TransportToken>
            <wsp:Policy>
              <sp:HttpsToken RequireClientCertificate="false" />
            </wsp:Policy>
          </sp:TransportToken>
          <sp:AlgorithmSuite>
            <wsp:Policy>
              <sp:Basic256 />
            </wsp:Policy>
          </sp:AlgorithmSuite>
          <sp:Layout>
            <wsp:Policy>
              <sp:Strict />
            </wsp:Policy>
          </sp:Layout>
          <sp:IncludeTimestamp />
        </wsp:Policy>
      </sp:TransportBinding>
      <sp:SignedSupportingTokens xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy>
          <sp:UsernameToken
sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipi
ent">
            <wsp:Policy>
              <sp:WssUsernameToken10 />
            </wsp:Policy>
          </sp:UsernameToken>
        </wsp:Policy>
      </sp:SignedSupportingTokens>
      <sp:Wss11 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy>
          <sp:MustSupportRefKeyIdentifier />
          <sp:MustSupportRefIssuerSerial />
          <sp:MustSupportRefThumbprint />
          <sp:MustSupportRefEncryptedKey />
        </wsp:Policy>
      </sp:Wss11>
      <sp:Trust10 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy>
          <sp:MustSupportIssuedTokens />
```

```
          <sp:RequireClientEntropy />
          <sp:RequireServerEntropy />
          </wsp:Policy>
      </sp:Trust10>
      <wsaw:UsingAddressing />
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsdl:import namespace="http://tempuri.org/"
location="https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/mex?wsdl=wsdl0"
/>
<wsdl:types>
  <xsd:schema
targetNamespace="http://schemas.microsoft.com/ws/2008/06/identity/securitytokenservice/Imports">
    <xsd:import
schemaLocation="https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/mex?xsd=x
sd0" namespace="http://schemas.microsoft.com/Message" />
  </xsd:schema>
</wsdl:types>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Cancel_InputMessage">
  <wsdl:part name="message" type="q1:MessageBody" xmlns:q1="http://schemas.microsoft.com/Message"
/>
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Cancel_OutputMessage">
  <wsdl:part name="TrustFeb2005CancelResult" type="q2:MessageBody"
xmlns:q2="http://schemas.microsoft.com/Message" />
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Issue_InputMessage">
  <wsdl:part name="message" type="q3:MessageBody" xmlns:q3="http://schemas.microsoft.com/Message"
/>
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Issue_OutputMessage">
  <wsdl:part name="TrustFeb2005IssueResult" type="q4:MessageBody"
xmlns:q4="http://schemas.microsoft.com/Message" />
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Renew_InputMessage">
  <wsdl:part name="message" type="q5:MessageBody" xmlns:q5="http://schemas.microsoft.com/Message"
/>
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Renew_OutputMessage">
  <wsdl:part name="TrustFeb2005RenewResult" type="q6:MessageBody"
xmlns:q6="http://schemas.microsoft.com/Message" />
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Validate_InputMessage">
  <wsdl:part name="message" type="q7:MessageBody" xmlns:q7="http://schemas.microsoft.com/Message"
/>
</wsdl:message>
<wsdl:message name="IWSTrustFeb2005Sync_TrustFeb2005Validate_OutputMessage">
  <wsdl:part name="TrustFeb2005ValidateResult" type="q8:MessageBody"
xmlns:q8="http://schemas.microsoft.com/Message" />
</wsdl:message>
<wsdl:portType name="IWSTrustFeb2005Sync">
  <wsdl:operation name="TrustFeb2005Cancel">
    <wsdl:input wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Cancel"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Cancel_InputMessage" />
    <wsdl:output wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Cancel"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Cancel_OutputMessage" />
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Issue">
    <wsdl:input wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Issue"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Issue_InputMessage" />
    <wsdl:output wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Issue"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Issue_OutputMessage" />
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Renew">
    <wsdl:input wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Renew"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Renew_InputMessage" />
    <wsdl:output wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Renew"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Renew_OutputMessage" />
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Validate">
```

```
    <wsdl:input wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Validate"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Validate_InputMessage" />
    <wsdl:output wsaw:Action="http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Validate"
message="tns:IWSTrustFeb2005Sync_TrustFeb2005Validate_OutputMessage" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="transportBindingConfig_IWSTrustFeb2005Sync" type="tns:IWSTrustFeb2005Sync">
  <wsp:PolicyReference URI="#transportBindingConfig_IWSTrustFeb2005Sync_policy" />
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="TrustFeb2005Cancel">
    <soap12:operation soapAction="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Cancel"
style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Issue">
    <soap12:operation soapAction="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Issue"
style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Renew">
    <soap12:operation soapAction="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Renew"
style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="TrustFeb2005Validate">
    <soap12:operation soapAction="http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Validate"
style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="SecurityTokenService">
  <wsdl:port name="transportBindingConfig_IWSTrustFeb2005Sync"
binding="tns:transportBindingConfig_IWSTrustFeb2005Sync">
    <soap12:address
location="https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts" />
    <wsa10:EndpointReference>

<wsa10:Address>https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts</wsa10
:Address>
      <Identity xmlns="http://schemas.xmlsoap.org/ws/2006/02/addressingidentity">
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <X509Data>
            <X509Certificate>MIIF+jCCBO ... ==</X509Certificate>
          </X509Data>
        </KeyInfo>
      </Identity>
    </wsa10:EndpointReference>
  </wsdl:port>
  <wsdl:port name="WSHttpBinding_IWSTrustFeb2005Sync"
binding="i0:WSHttpBinding_IWSTrustFeb2005Sync">
```

```
    <soap12:address
location="https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts/x509" />
    <wsa10:EndpointReference>

<wsa10:Address>https://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts/x509</
wsa10:Address>
      <Identity xmlns="http://schemas.xmlsoap.org/ws/2006/02/addressingidentity">
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <X509Data>
            <X509Certificate>MIIF+jCCBO ... ==</X509Certificate>
          </X509Data>
        </KeyInfo>
      </Identity>
    </wsa10:EndpointReference>
  </wsdl:port>
  <wsdl:port name="WSFederationHttpBinding_IWSTrustFeb2005Sync"
binding="i0:WSFederationHttpBinding_IWSTrustFeb2005Sync">
    <soap12:address
location="http://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts/self" />
    <wsa10:EndpointReference>

<wsa10:Address>http://ipsts.federatedidentity.net/SecurityTokenService/InteropSts.svc/Sts/self</w
sa10:Address>
      <Identity xmlns="http://schemas.xmlsoap.org/ws/2006/02/addressingidentity">
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <X509Data>
            <X509Certificate>MIIF+jCCBO ... ==</X509Certificate>
          </X509Data>
        </KeyInfo>
      </Identity>
    </wsa10:EndpointReference>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

Note that this example illustrates that WSDL documents may refer to other documents, which themselves need to also be retrieved. These separate WSDL documents are referenced via the wsdl:import statements in the base document.

Appendix B – Windows CardSpace .NET Framework 3.5 Service Pack 1 Constraints

The Identity Selector Interoperability Profile V1.5 was used to implement the Windows CardSpace software in Microsoft .NET Framework 3.5 Service Pack 1. This section documents any additional constraints imposed by the Windows CardSpace .NET Framework 3.5 Service Pack 1 implementation or where it differs from the V1.5 profile. All references to section numbers below are with respect to the [ISIP] profile document.

- In reference to Section 3.1.1, when retrieving the WSDL of an RP/STS, Windows CardSpace will first attempt to retrieve the WSDL document via [WS-MetadataExchange] and then HTTPS GET.
- In reference to Section 4.1.1.2, when retrieving the WSDL of an IP/STS, Windows CardSpace will first attempt to retrieve the WSDL document via [WS-MetadataExchange] and then HTTPS GET.