[MS-CPSWS]: SharePoint Claim Provider Web Service Protocol Specification

Release: Sunday, December 19, 2010

Revision Summary

Date	Revision History	Revision Class	Comments
07/13/2009	0.1	Major	Initial Availability
08/28/2009	0.2	Editorial	Revised and edited the technical content
11/06/2009	0.3	Editorial	Revised and edited the technical content
02/19/2010	1.0	Major	Updated and revised the technical content
03/31/2010	1.01	Editorial	Revised and edited the technical content
04/30/2010	1.02	Editorial	Revised and edited the technical content
06/07/2010	1.03	Editorial	Revised and edited the technical content
06/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	1.05	Minor	Clarified the meaning of the technical content.
09/27/2010	1.05	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.05	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	1.05	No change	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

This document specifies the SharePoint Claim Provider Web Service Protocol. This protocol enables a protocol client to access claims, which are a set of statements one subject makes about itself or another subject. Claims are returned by the claims providers that are available on a protocol server. A typical scenario for this protocol is a client application that allows users to search and resolve users and groups from a set of directory services and external identity providers.

1.1 Glossary

The following terms are defined in [MS-GLOS]:

directory service (DS)
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)

The following terms are defined in [MS-OFCGLOS]:

claim
claim type
claim value
claim-based authentication mode
claims provider
claims provider schema
hierarchy provider
SOAP (Simple Object Access Protocol)
SOAP action
SOAP body
SOAP fault
Status-Code
URI (Uniform Resource Identifier)
Web application
WSDL (Web Services Description Language)
WSDL message
WSDL operation
XML namespace
XML Schema

The following terms are specific to this document:

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.ietf.org/rfc/rfc2119.txt

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, http://www.ietf.org/rfc/rfc2616.txt

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, http://www.w3.org/TR/2003/REC-soap12-part1-20030624

[WSA1.0 Metadata] Gudgin, M., Ed., Hadley, M., Ed., Rogers, T., Ed., Yalcinalp, U., Ed., "Web Services Addressing 1.0 - Metadata", W3C Recommendation 4 September 2007, http://www.w3.org/TR/ws-addr-metadata/

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, http://www.w3.org/TR/2001/NOTE-wsdl-20010315

[XMLNS] World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation 8 December 2009, http://www.w3.org/TR/REC-xml-names/

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary", March 2007.

[MS-OFCGLOS] Microsoft Corporation, "Microsoft Office Master Glossary", June 2008.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, http://www.ietf.org/rfc/rfc2818.txt

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, http://www.w3.org/TR/2003/REC-soap12-part2-20030624

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to access **claims (2)** returned by the **claims providers** available for a specific **Web application (2)**. A typical scenario for this protocol is a client application that allows users to browse, search, and resolve people and groups from a set of **directory services** and external identity providers.

This protocol allows a protocol client to send a search query of claims providers and obtain information necessary to store a reference to each claim.

Claims providers can also return metadata relevant to each claim and return claims in hierarchical trees for better presentation by the protocol client.

1.4 Relationship to Other Protocols

This protocol uses the **SOAP** message protocol for formatting request and response messages, as described in [SOAP1.1], [SOAP1.2/1] and [SOAP1.2/2]. It transmits those messages by using **HTTP**, as described in [RFC2616], or **HTTPS**, as described in [RFC2818].

The following diagram shows the underlying messaging and transport stack used by the protocol:

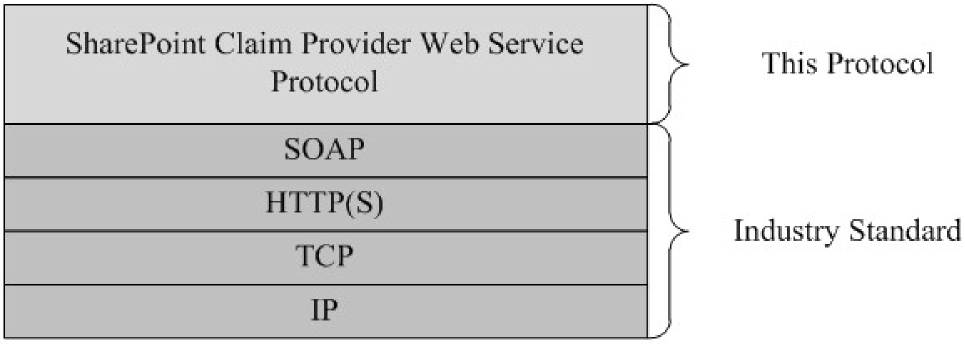

Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a site that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "/_vti_bin/SPClaimProviderWebService.svc" to the URL of the site, for example: http://www.contoso.com/_vti_bin/SPClaimProviderWebService.svc.

This protocol assumes that authentication has been performed by the underlying protocols.

1.6 Applicability Statement

This protocol is applicable when a client application allows users to browse, search, and resolve people and groups from a set of directory services and external identity providers through claims providers.

This protocol does not specify how the claims providers interface with the directory services or identity providers or how that data is stored.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

In the following sections, the schema definition might be less restrictive than the processing rules imposed by the protocol. The **WSDL** in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL specifies additional restrictions that reflect protocol behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **present**, and **not null**.

2.1 Transport

Protocol servers MUST support SOAP over HTTP. Protocol servers SHOULD additionally support SOAP over HTTPS for securing communication with protocol clients.

Protocol messages MUST be formatted as specified in [SOAP1.1], section 4 or in [SOAP1.2/1], section 5. Protocol server faults MUST be returned using HTTP **Status-Codes** as specified in [RFC2616], section 10 or using **SOAP faults** as specified in [SOAP1.1], section 4.4 or [SOAP1.2/1], section 5.4.

2.2 Common Message Syntax

This section contains common structures used by this protocol. The syntax of the structures uses **XML Schema**, as specified in [XMLSCHEMA1] and [XMLSCHEMA2], and Web Services Description Language, as specified in [WSDL].

2.2.1 Namespaces

This protocol specifies and references **XML namespaces** using the mechanisms specified in [XMLNS]. Although this document associates an XML namespace prefix for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://microsoft.com/webservices/SharePointPortalServer/UserProfileService	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
xsd	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]
tns	http://schemas.microsoft.com/office/Excel/Server/WebServices/ExcelServerInternalService/	
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]
wsam	http://www.w3.org/2007/05/addressing/metadata	[WSA1.0 Metadata]

2.2.2 Messages

None.

2.2.3 Elements

This specification does not define any common XML Schema element definitions.

2.2.4 Complex Types

The following table summarizes the set of common XML Schema complex type definitions defined by this specification. XML Schema complex type definitions that are specific to a particular operation are described with the operation.

Complex Type	Description
ArrayOfAnyType	An array of elements of any type.
ArrayOfPair	An array of elements of type **Pair** (section 2.2.4.8).
ArrayOfPickerEntity	An array of elements of type **PickerEntity** (section 2.2.4.9).
ArrayOfSPProviderHierarchyNode	An array of elements of type **SPProviderHierarchyNode** (section 2.2.4.12).
ArrayOfSPProviderHierarchyTree	An array of elements of type **SPProviderHierarchyTree** (section 2.2.4.13).
ArrayOfSPSchemaElement	An array of **SPSchemaElement** (section 2.2.4.15) elements.
ArrayOfString	An array of elements of type string.
Pair	A collection of two named elements of any type in a sequence.
PickerEntity	An object containing basic information about a picker entity.
SPClaim	The claim (2) associated with a login entity.
SPProviderHierarchyElement	Defines the base type for **SPProviderHierarchyNode** and **SPProviderHierarchyTree**.
SPProviderHierarchyNode	A claims provider hierarchy node.
SPProviderHierarchyTree	A claims provider hierarchy tree.
SPProviderSchema	The user interface display characteristics of a claims provider.
SPSchemaElement	The user interface display characteristics of a field in a picker entity.

2.2.4.1 ArrayOfAnyType

An array of elements of any type.

```
<xs:complexType name="ArrayOfAnyType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true"/>
  </xs:sequence>
```

```
</xs:complexType>
```

anyType: An anyType value as specified in [XMLSCHEMA1].

2.2.4.2 ArrayOfPair

An array of elements of type **Pair** (section 2.2.4.8).

```
<xs:complexType name="ArrayOfPair">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Pair" nillable="true" 
type="tns:Pair"/>
  </xs:sequence>
</xs:complexType>
```

Pair: A collection of two elements.

2.2.4.3 ArrayOfPickerEntity

An array of elements of type **PickerEntity** (section 2.2.4.9).

```
<xs:complexType name="ArrayOfPickerEntity">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="PickerEntity" nillable="true" 
type="tns:PickerEntity"/>
  </xs:sequence>
</xs:complexType>
```

PickerEntity: An element containing information about a picker entity.

2.2.4.4 ArrayOfSPProviderHierarchyNode

An array of elements of type **SPProviderHierarchyNode** (section 2.2.4.12).

```
<xs:complexType name="ArrayOfSPProviderHierarchyNode">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderHierarchyNode" 
nillable="true" type="tns:SPProviderHierarchyNode"/>
  </xs:sequence>
</xs:complexType>
```

SPProviderHierarchyNode: An element containing information about a provider hierarchy node.

2.2.4.5 ArrayOfSPProviderHierarchyTree

An array of elements of type **SPProviderHierarchyTree** (section 2.2.4.13).

```
<xs:complexType name="ArrayOfSPProviderHierarchyTree">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderHierarchyTree" 
nillable="true" type="tns:SPProviderHierarchyTree"/>
  </xs:sequence>
```

```
</xs:complexType>
```

SPProviderHierarchyTree: An element containing information about a claims provider hierarchy tree.

2.2.4.6 ArrayOfSPSchemaElement

An array of **SPSchemaElement** (section 2.2.4.15) elements.

```
<xs:complexType name="ArrayOfSPSchemaElement">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPSchemaElement" nillable="true"
type="tns:SPSchemaElement"/>
  </xs:sequence>
</xs:complexType>
```

SPSchemaElement: The user interface display characteristics of a field in a login entity.

2.2.4.7 ArrayOfString

An array of elements of type string.

```
<xs:complexType name="ArrayOfString">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

string: A string as specified in [XMLSCHEMA1].

2.2.4.8 Pair

A collection of two named elements of any type in a sequence.

```
<xs:complexType name="Pair">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="First"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Second"/>
  </xs:sequence>
</xs:complexType>
```

First: The element at the first position.

Second: The element at the second position.

2.2.4.9 PickerEntity

An object containing basic information about a picker entity.

```
<xs:complexType name="PickerEntity">
  <xs:sequence>
```

```
      <xs:element minOccurs="0" maxOccurs="1" name="Key" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="DisplayText" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="IsResolved" type="xs:boolean"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityGroupName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="HierarchyIdentifier"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityDataElements"
  type="tns:ArrayOfPair"/>
      <xs:element minOccurs="0" maxOccurs="1" name="MultipleMatches"
  type="tns:ArrayOfAnyType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
```

Key: The name that uniquely identifies the picker entity.

DisplayText: The name to be used in the user interface.

IsResolved: A Boolean value that indicates if the picker entity has been successfully resolved to a unique match.

Description: The detailed description for the picker entity.

EntityType: The type for the picker entity.

EntityGroupName: The group name of the picker entity.

HierarchyIdentifier: The identifier of the picker entity in the hierarchy.

EntityDataElements: The list of name-value pairs representing the picker entity data.

MultipleMatches: The list of matched picker entities when there is more than one match.

ProviderName: The name of a claims provider.

2.2.4.10 SPClaim

The claim (2) associated with a login entity.

```
  <xs:complexType name="SPClaim">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ClaimType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="ValueType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="OriginalIssuer" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
```

ClaimType: The **claim type**.

Value: The **claim value**.

ValueType: The type of the claim value.

OriginalIssuer: The original issuer of the claim (2). The original issuer string MUST begin with one of the following strings representing type of the issuer followed by an optional valid identifier of the issuer. The original issuer type and identifier MUST be separated by a single colon (:) character.

Prefix	Comment
Windows	Issuer is of type Windows.
SecurityTokenService	Issuer is of type Security Token Service.
Forms	Issuer is of type Forms. The protocol client MUST specify valid identifier of the issuer.
TrustedProvider	Issuer is of type Trusted Provider. The protocol client MUST specify valid identifier of the issuer.
ClaimsProvider	Issuer is of type Claims Provider. The protocol client MUST specify valid identifier of the issuer.

A valid identifier MUST not begin or end with spaces, MUST not contain any control character or character that can be used as XML markup, and MUST not contain any of the following characters: colon (:), comma (,) or semicolon (;).

2.2.4.11 SPProviderHierarchyElement

Defines the base type for **SPProviderHierarchyNode** and **SPProviderHierarchyTree**.

```
<xs:complexType name="SPProviderHierarchyElement">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Nm" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="IsLeaf" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Children" 
type="tns:ArrayOfSPProviderHierarchyNode"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityData" 
type="tns:ArrayOfPickerEntity"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Nm: The human-readable display name.

ProviderName: The name of the claims provider.

HierarchyNodeID: The unique identifier for the element.

IsLeaf: Whether or not the element is a leaf element. This value is true if and only if the element has no subelements. Otherwise, it is false.

Children: The list of immediate child elements.

EntityData: The **PickerEntity** objects associated with this element.

Count: The number of **PickerEntity** objects in this element and all child elements, calculated recursively.

2.2.4.12 SPProviderHierarchyNode

A claims provider hierarchy node.

```
<xs:complexType name="SPProviderHierarchyNode">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPProviderHierarchyElement"/>
  </xs:complexContent>
</xs:complexType>
```

2.2.4.13 SPProviderHierarchyTree

A claims provider hierarchy tree.

```
<xs:complexType name="SPProviderHierarchyTree">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPProviderHierarchyElement">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="IsRoot" type="xs:boolean"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

IsRoot: A Boolean value that indicates whether the node root of this tree is also root of the claims provider’s complete hierarchy tree.

2.2.4.14 SPProviderSchema

The user interface display characteristics of a claims provider.

```
<xs:complexType name="SPProviderSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchema" 
type="tns:ArrayOfSPSchemaElement"/>
    <xs:element minOccurs="1" maxOccurs="1" name="SupportsHierarchy" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

DisplayName: The name to be used in user interfaces.

ProviderName: The name of a claims provider.

ProviderSchema: The list of schema elements that comprise the claims provider schema.

SupportsHierarchy: A Boolean value that indicates whether the claims provider supports hierarchy.

2.2.4.15 SPSchemaElement

The user interface display characteristics of a field in a picker entity.

```
<xs:complexType name="SPSchemaElement">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SPSchemaElementType"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name that uniquely identifies the field.

DisplayName: The name to be used for the field in user interfaces.

Type: The display type of a field in a picker entity.

2.2.5 Simple Types

The following table summarizes the set of common XML Schema simple type definitions defined by this specification. XML Schema simple type definitions that are specific to a particular operation are described with the operation.

Simple Type	Description
SPPrincipalType	Contains the value that is used to specify the type(s) of picker entities included in the output claims provider hierarchy tree.
SPSchemaElementType	The display type of a field.

2.2.5.1 SPPrincipalType

Contains the value that is used to specify the type(s) of picker entities included in the output claims provider hierarchy tree.

```
<xs:simpleType name="SPPrincipalType">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="User"/>
        <xs:enumeration value="DistributionList"/>
        <xs:enumeration value="SecurityGroup"/>
        <xs:enumeration value="SharePointGroup"/>
        <xs:enumeration value="All"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

The following table specifies the allowable values for SPPrincipalType:

Value	Meaning
None	No picker entity is included in the output claims provider hierarchy tree.
User	Only picker entities of type user are included in the output claims provider hierarchy

Value	Meaning
	tree.
DistributionList	Only picker entities of type Distribution List are included in the output claims provider hierarchy tree.
SecurityGroup	Picker entities of type Security Group and all other types except User and Distribution List are included in the output claims provider hierarchy tree.
SharePointGroup	No picker entity is included in the output claims provider hierarchy tree.
All	No picker entity is included in the output claims provider hierarchy tree.

2.2.5.2 SPSchemaElementType

The display type of a field.

```
<xs:simpleType name="SPSchemaElementType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None"/>
    <xs:enumeration value="TableViewOnly"/>
    <xs:enumeration value="DetailViewOnly"/>
    <xs:enumeration value="Both"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for SPSchemaElementType:

Value	Meaning
None	The field is not intended to be displayed in the user interface.
TableViewOnly	The field to be displayed in the Table-View only.
DetailViewOnly	The field to be displayed in the Detailed-View only.
Both	The field can be displayed either in Table-View or Detailed-View.

2.2.6 Attributes

This specification does not define any common XML Schema attribute definitions.

2.2.7 Groups

This specification does not define any common XML Schema group definitions.

2.2.8 Attribute Groups

This specification does not define any common XML Schema attribute group definitions.

3 Protocol Details

In the following sections, the schema definition might be less restrictive than the processing rules imposed by the protocol. The WSDL in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL specifies additional restrictions that reflect protocol behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **present**, and **not null**.

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

Except where specified, protocol clients SHOULD interpret HTTP Status-Codes returned by the protocol server as specified in [RFC2616], section 10, Status-Code Definitions.

This protocol allows protocol servers to notify protocol clients of application-level faults using SOAP faults. Except where specified, these SOAP faults are not significant for interoperability, and protocol clients can interpret them in an implementation-specific manner.

This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults either using HTTP Status-Codes or using SOAP faults as specified previously in this section.

3.1 Server Details

All operations consist of a basic request-response pair and the protocol server treats each request as an independent transaction that is unrelated to any previous request.

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

The protocol server maintains the following data:

- A list of Web applications (2) in **claim-based authentication mode**.
- A list of claims providers available to the protocol client.

Each claims provider maintains the following data:

- A list of claims (2) that the protocol client can search or resolve against. This protocol does not enforce any implementation on how the list of claims is maintained or the search and resolve operations determined by each claims provider.
- A claims provider hierarchy tree: The tree structure representing the hierarchy of claims as determined by the claim provider. For example, an organization tree could contain a top-level "Contoso" node, a "Sales" child node with 23 claims and an "Engineering" node with 18 claims.
- A **claims provider schema**.

Each claim (2) maintains the following data:

- A claim value.
- A claim type.
- A claim value type.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

There are no special sequencing rules for the operations defined in this specification.

This specification includes the following **WSDL operations**:

WSDL Operation	Description
ClaimTypes	This operation is used to retrieve a list of all possible claim types from a list of claims providers.
ClaimValueTypes	This operation is used to retrieve a list of claim value types from a list of claims providers.
EntityTypes	This operation is used to retrieve a list of all possible picker entity types from a list of claims providers.
GetHierarchy	This operation is used to retrieve a claims provider hierarchy tree from a claims provider.
GetHierarchyAll	This operation is used to retrieve a list of claims provider hierarchy trees from a list of claims providers.
HierarchyProviderSchema	This operation is used to retrieve schema for the current **hierarchy provider**.
ProviderSchemas	This operation is used to retrieve a list of claims provider schemas from a list of claims providers.
Resolve	This operation is used to resolve an input string to picker entities using a list of claims providers.
ResolveClaim	This operation is used to resolve an input claim (2) to picker entities using a list of claims providers.
ResolveMultiple	This operation is used to resolve a list of strings to picker entities using a list of claims providers.
ResolveMultipleClaim	This operation is used to resolve a list of claims to picker entities using a list of claims providers.
Search	This operation is used to perform a search for entities on a list of claims providers. The input to this operation is a search string and the result of this operation is

WSDL Operation	Description
	a list of claims provider hierarchy trees that contain the matched entities. If a hierarchy provider is available, the protocol server MAY move results returned by claims providers to hierarchy provider hierarchy tree.
SearchAll	This operation is used to perform a search for entities on a list of claims providers. The result of this operation is a list of claims provider hierarchy trees. If a hierarchy provider is available, the protocol server MAY move entities returned by claims providers to hierarchy provider hierarchy tree.

3.1.4.1 ClaimTypes

This operation is used to retrieve a list of all possible claim types from a list of claims providers.

```
<wsdl:operation name="ClaimTypes">
  <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTyp
es" message="tns:IClaimProviderWebService_ClaimTypes_InputMessage"/>
  <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTyp
esResponse" message="tns:IClaimProviderWebService_ClaimTypes_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ClaimTypes_InputMessage** (section 3.1.4.1.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ClaimTypes_OutputMessage** (section 3.1.4.1.1.2) response message.

The protocol server MUST retrieve all known basic claim types. In addition, the protocol server MUST retrieve claim types from claims providers that meet both of the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.

If the list of provider names is NULL, the protocol server MUST use all the available claims providers.

3.1.4.1.1 Messages

The following **WSDL message** definitions are specific to this operation.

3.1.4.1.1.1 IClaimProviderWebService_ClaimTypes_InputMessage

The request WSDL message for the **ClaimTypes** WSDL operation.

The **SOAP action** value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypes
```

The **SOAP body** contains the **ClaimTypes** element.

3.1.4.1.1.2 IClaimProviderWebService_ClaimTypes_OutputMessage

The response WSDL message for the **ClaimTypes** WSDL operation.

The SOAP body contains the **ClaimTypesResponse** element.

3.1.4.1.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.1.2.1 ClaimTypes

The input data for the **ClaimTypes** WSDL operation.

```
<xs:element name="ClaimTypes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names.

3.1.4.1.2.2 ClaimTypesResponse

The result data for the **ClaimTypes** WSDL operation.

```
<xs:element name="ClaimTypesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ClaimTypesResult" 
type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ClaimTypesResult: The list of claim types supported by the claims providers.

3.1.4.1.3 Complex Types

None.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 ClaimValueTypes

This operation is used to retrieve a list of claim value types from a list of claims providers.

```
<wsdl:operation name="ClaimValueTypes">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimVal
ueTypes" message="tns:IClaimProviderWebService_ClaimValueTypes_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimVal
ueTypesResponse" message="tns:IClaimProviderWebService_ClaimValueTypes_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ClaimValueTypes_InputMessage** (section 3.1.4.2.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ClaimValueTypes_OutputMessage** (section 3.1.4.2.1.2) response message.

The protocol server MUST retrieve all known basic claim value types. In addition, the protocol server MUST retrieve claim value types from claims providers that meet both of the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.2.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.2.1.1 IClaimProviderWebService_ClaimValueTypes_InputMessage

The request WSDL message for the **ClaimValueTypes** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypes
```

The SOAP body contains the **ClaimValueTypes** element.

3.1.4.2.1.2 IClaimProviderWebService_ClaimValueTypes_OutputMessage

The response WSDL message for the **ClaimValueTypes** WSDL operation.

The SOAP body contains the **ClaimValueTypesResponse** element.

3.1.4.2.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.2.2.1 ClaimValueTypes

The input data for the **ClaimValueTypes** WSDL operation.

```
<xs:element name="ClaimValueTypes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names.

3.1.4.2.2.2 ClaimValueTypesResponse

The result data for the **ClaimValueTypes** WSDL operation.

```
<xs:element name="ClaimValueTypesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ClaimValueTypesResult" 
type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ClaimValueTypesResult: The list of claim value types.

3.1.4.2.3 Complex Types

None.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 EntityTypes

This operation is used to retrieve a list of all possible picker entity types from a list of claims providers.

```
<wsdl:operation name="EntityTypes">
  <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTy
pes" message="tns:IClaimProviderWebService_EntityTypes_InputMessage"/>
```

```
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTy
pesResponse" message="tns:IClaimProviderWebService_EntityTypes_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_EntityTypes_InputMessage** (section 3.1.4.3.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_EntityTypes_OutputMessage** (section 3.1.4.3.1.2) response message.

The protocol server MUST retrieve picker entity types from **claims providers** that meet both of the following criteria:

- The claims providers are associated with the Web application specified in the input message.
- The claims providers are listed in the provider names in the input message.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.3.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.3.1.1 IClaimProviderWebService_EntityTypes_InputMessage

The request WSDL message for the **EntityTypes** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypes
```

The SOAP body contains the **EntityTypes** element.

3.1.4.3.1.2 IClaimProviderWebService_EntityTypes_OutputMessage

The response WSDL message for the **EntityTypes** WSDL operation.

The SOAP body contains the **EntityTypesResponse** element.

3.1.4.3.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.3.2.1 EntityTypes

The input data for the **EntityTypes** WSDL operation.

```
<xs:element name="EntityTypes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

3.1.4.3.2.2 EntityTypesResponse

The result data for the **EntityTypes** WSDL operation.

```
<xs:element name="EntityTypesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityTypesResult" 
type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

EntityTypesResult: The list of picker entity types supported by the claims providers.

3.1.4.3.3 Complex Types

None.

3.1.4.3.4 Simple Types

None.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.4.4 GetHierarchy

This operation is used to retrieve a claims provider hierarchy tree from a claims provider.

```
<wsdl:operation name="GetHierarchy">
  <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchy" message="tns:IClaimProviderWebService_GetHierarchy_InputMessage"/>
  <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyResponse" message="tns:IClaimProviderWebService_GetHierarchy_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_GetHierarchy_InputMessage** (section 3.1.4.4.1.1) request WSDL message and the protocol server responds with an

IClaimProviderWebService_GetHierarchy_OutputMessage (section 3.1.4.4.1.2) response message.

The protocol server MUST retrieve a claims provider hierarchy tree from the claims provider that meets all the following criteria:

- The claims provider name is specified in the input message.
- The claims provider is associated with the Web application (2) specified in the input message.
- The claims provider supports hierarchy.

3.1.4.4.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.4.1.1 IClaimProviderWebService_GetHierarchy_InputMessage

The request WSDL message for the **GetHierarchy** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchy
```

The SOAP body contains the **GetHierarchy** element.

3.1.4.4.1.2 IClaimProviderWebService_GetHierarchy_OutputMessage

The response WSDL message for the **GetHierarchy** WSDL operation.

The SOAP body contains the **GetHierarchyResponse** element.

3.1.4.4.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.4.2.1 GetHierarchy

The input data for the **GetHierarchy** WSDL operation.

```
<xs:element name="GetHierarchy">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="hierarchyNodeID" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerName: The claims provider name. The value of this element MUST NOT be NULL.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

hierarchyNodeID: Identifier of the node to be used as root of the returned claims provider hierarchy tree. If NULL is specified, then the protocol server MUST return the existing root of claims provider hierarchy tree.

numberOfLevels: The maximum number of levels that can be returned by the protocol server in any of the output claims provider hierarchy tree. The value of this element MUST NOT be less than 1.

The protocol server MUST return all levels up to the specified number of levels from the root of the claims provider hierarchy tree. If a claims provider hierarchy tree has more than specified number of levels, then the protocol server MUST only return the specified number of levels from the root of the claims provider hierarchy tree.

The protocol server MUST return **ArgumentOutOfRangeException<"numberOfLevels">** message if the value of this element is less than 1.

3.1.4.4.2.2 GetHierarchyResponse

The result data for the **GetHierarchy** WSDL operation.

```
<xs:element name="GetHierarchyResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyResult" 
type="tns:SPProviderHierarchyTree"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetHierarchyResult: A claims provider hierarchy tree.

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 GetHierarchyAll

This operation is used to retrieve a list of claims provider hierarchy trees from a list of claims providers.

```
<wsdl:operation name="GetHierarchyAll">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyAll" message="tns:IClaimProviderWebService_GetHierarchyAll_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyAllResponse" message="tns:IClaimProviderWebService_GetHierarchyAll_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an IClaimProviderWebService_GetHierarchyAll_InputMessage (section 3.1.4.5.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_GetHierarchyAll_OutputMessage** (section 3.1.4.5.1.2) response message.

The protocol server MUST retrieve claims provider hierarchy trees from claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support hierarchy.

The protocol server MUST also retrieve hierarchy provider hierarchy tree if one is available.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.5.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.5.1.1 IClaimProviderWebService_GetHierarchyAll_InputMessage

The request WSDL message for the **GetHierarchyAll** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAll
```

The SOAP body contains the **GetHierarchyAll** element.

3.1.4.5.1.2 IClaimProviderWebService_GetHierarchyAll_OutputMessage

The response WSDL message for the **GetHierarchyAll** WSDL operation.

The SOAP body contains the **GetHierarchyAllResponse** element.

3.1.4.5.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.5.2.1 GetHierarchyAll

The input data for the **GetHierarchyAll** WSDL operation.

```
<xs:element name="GetHierarchyAll">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return hierarchy tree for all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

numberOfLevels: The maximum number of levels that can be returned by the protocol server in any of the output claims provider hierarchy trees. The value of this element MUST NOT be less than 1.

The protocol server MUST return all levels up to the specified number of levels from the root of the claims provider hierarchy tree. If a claims provider hierarchy tree has more than the specified number of levels, then the protocol server MUST only return the specified number of levels from the root of the claims provider hierarchy tree.

The protocol server MUST return an **ArgumentOutOfRangeException<"numberOfLevels">** message if the value of this element is less than 1.

3.1.4.5.2.2 GetHierarchyAllResponse

The result data for the **GetHierarchyAll** WSDL operation.

```
<xs:element name="GetHierarchyAllResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyAllResult"
type="tns:ArrayOfSPProviderHierarchyTree"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetHierarchyAllResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that match the criteria specified in the input message.

3.1.4.5.3 Complex Types

None.

3.1.4.5.4 Simple Types

None.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.4.6 HierarchyProviderSchema

This operation is used to retrieve schema for the current hierarchy provider.

```
<wsdl:operation name="HierarchyProviderSchema">
  <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarch
yProviderSchema" 
message="tns:IClaimProviderWebService_HierarchyProviderSchema_InputMessage"/>
  <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarch
yProviderSchemaResponse" 
message="tns:IClaimProviderWebService_HierarchyProviderSchema_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_HierarchyProviderSchema_InputMessage** (section 3.1.4.6.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_HierarchyProviderSchema_OutputMessage** (section 3.1.4.6.1.2) response message.

3.1.4.6.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.6.1.1 IClaimProviderWebService_HierarchyProviderSchema_InputMessage

The request WSDL message for the **HierarchyProviderSchema** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/HierarchyProviderSche
ma
```

The SOAP body contains the **HierarchyProviderSchema** element.

3.1.4.6.1.2 IClaimProviderWebService_HierarchyProviderSchema_OutputMessage

The response WSDL message for the **HierarchyProviderSchema** WSDL operation.

The SOAP body contains the **HierarchyProviderSchemaResponse** element.

3.1.4.6.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.6.2.1 HierarchyProviderSchema

The input data for the **HierarchyProviderSchema** WSDL operation.

```
<xs:element name="HierarchyProviderSchema">
  <xs:complexType/>
</xs:element>
```

3.1.4.6.2.2 HierarchyProviderSchemaResponse

The result data for the **HierarchyProviderSchema** WSDL operation.

```
<xs:element name="HierarchyProviderSchemaResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="HierarchyProviderSchemaResult" 
type="tns:SPProviderSchema"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

HierarchyProviderSchemaResult: The list of hierarchy claims provider schemas.

3.1.4.6.3 Complex Types

None.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6 Groups

None.

3.1.4.6.7 Attribute Groups

None.

3.1.4.7 ProviderSchemas

This operation is used to retrieve a list of claims provider schemas from a list of claims providers.

```
<wsdl:operation name="ProviderSchemas">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Provider
Schemas" message="tns:IClaimProviderWebService_ProviderSchemas_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Provider
SchemasResponse" message="tns:IClaimProviderWebService_ProviderSchemas_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ProviderSchemas_InputMessage** (section 3.1.4.7.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ProviderSchemas_OutputMessage** (section 3.1.4.7.1.2) response message.

The protocol server MUST retrieve claims provider schemas from claims providers that meet all the following criteria:

- The claims providers are enabled for the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message or SharePoint Groups provider.
- The claims providers support search and resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.7.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.7.1.1 IClaimProviderWebService_ProviderSchemas_InputMessage

The request WSDL message for the **ProviderSchemas** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemas
```

The SOAP body contains the **ProviderSchemas** element.

3.1.4.7.1.2 IClaimProviderWebService_ProviderSchemas_OutputMessage

The response WSDL message for the **ProviderSchemas** WSDL operation.

The SOAP body contains the **ProviderSchemasResponse** element.

3.1.4.7.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.7.2.1 ProviderSchemas

The input data for the **ProviderSchemas** WSDL operation.

```
<xs:element name="ProviderSchemas">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

3.1.4.7.2.2 ProviderSchemasResponse

The result data for the **ProviderSchemas** WSDL operation.

```
<xs:element name="ProviderSchemasResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchemasResult" 
type="tns:ArrayOfSPProviderSchema"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ProviderSchemasResult: The list of claims provider schemas.

3.1.4.7.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.7.3.1 ArrayOfSPProviderSchema

An array of **SPProviderSchema** (section 2.2.4.14) elements.

```
<xs:complexType name="ArrayOfSPProviderSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderSchema" nillable="true" 
type="tns:SPProviderSchema"/>
  </xs:sequence>
</xs:complexType>
```

SPProviderSchema: The user interface display characteristics of a field in an entity.

3.1.4.7.4 Simple Types

None.

3.1.4.7.5 Attributes

None.

3.1.4.7.6 Groups

None.

3.1.4.7.7 Attribute Groups

None.

3.1.4.8 Resolve

This operation is used to resolve an input string to picker entities using a list of claims providers.

```
<wsdl:operation name="Resolve">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve"
message="tns:IClaimProviderWebService_Resolve_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveR
esponse" message="tns:IClaimProviderWebService_Resolve_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_Resolve_InputMessage** (section 3.1.4.8.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_Resolve_OutputMessage** (section 3.1.4.8.1.2) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.8.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.8.1.1 IClaimProviderWebService_Resolve_InputMessage

The request WSDL message for the **Resolve** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve
```

The SOAP body contains the **Resolve** element.

3.1.4.8.1.2 IClaimProviderWebService_Resolve_OutputMessage

The response WSDL message for the **Resolve** WSDL operation.

The SOAP body contains the **ResolveResponse** element.

3.1.4.8.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.8.2.1 Resolve

The input data for the **Resolve** WSDL operation.

```
<xs:element name="Resolve">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The input to be resolved. This MUST NOT be NULL. If this is NULL, the protocol server MUST return an **ArgumentNullException<"resolveInput">** message.

3.1.4.8.2.2 ResolveResponse

The result data for the **Resolve** WSDL operation.

```
<xs:element name="ResolveResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveResult" 
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResolveResult: The list of picker entities.

3.1.4.8.3 Complex Types

None.

3.1.4.8.4 Simple Types

None.

3.1.4.8.5 Attributes

None.

3.1.4.8.6 Groups

None.

3.1.4.8.7 Attribute Groups

None.

3.1.4.9 ResolveClaim

This operation is used to resolve an input claim (2) to picker entities using a list of claims providers.

```
<wsdl:operation name="ResolveClaim">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laim" message="tns:IClaimProviderWebService_ResolveClaim_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laimResponse" message="tns:IClaimProviderWebService_ResolveClaim_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ResolveClaim_InputMessage** (section 3.1.4.9.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveClaim_OutputMessage** (section 3.1.4.9.1.2) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.9.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.9.1.1 IClaimProviderWebService_ResolveClaim_InputMessage

The request WSDL message for the **ResolveClaim** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveClaim
```

The SOAP body contains the **ResolveClaim** element.

3.1.4.9.1.2 IClaimProviderWebService_ResolveClaim_OutputMessage

The response WSDL message for the **ResolveClaim** WSDL operation.

The SOAP body contains the **ResolveClaimResponse** element.

3.1.4.9.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.9.2.1 ResolveClaim

The input data for the **ResolveClaim** WSDL operation.

```
<xs:element name="ResolveClaim">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:SPClaim"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The input to be resolved, represented as an **SPClaim** (section 2.2.4.10). The value of this element MUST NOT be NULL.

3.1.4.9.2.2 ResolveClaimResponse

The result data for the **ResolveClaim** WSDL operation.

```
<xs:element name="ResolveClaimResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveClaimResult" 
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResolveClaimResult: The list of picker entities.

3.1.4.9.3 Complex Types

None.

3.1.4.9.4 Simple Types

None.

3.1.4.9.5 Attributes

None.

3.1.4.9.6 Groups

None.

3.1.4.9.7 Attribute Groups

None.

3.1.4.10 ResolveMultiple

This operation is used to resolve a list of strings to picker entities using a list of claims providers.

```
<wsdl:operation name="ResolveMultiple">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultiple" message="tns:IClaimProviderWebService_ResolveMultiple_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleResponse" message="tns:IClaimProviderWebService_ResolveMultiple_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an IClaimProviderWebService_ResolveMultiple_InputMessage (section 3.1.4.10.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveMultiple_OutputMessage** (section 3.1.4.10.1.2) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.10.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.10.1.1 IClaimProviderWebService_ResolveMultiple_InputMessage

The request WSDL message for the **ResolveMultiple** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultiple
```

The SOAP body contains the **ResolveMultiple** element.

3.1.4.10.1.2 IClaimProviderWebService_ResolveMultiple_OutputMessage

The response WSDL message for the **ResolveMultiple** WSDL operation.

The SOAP body contains the **ResolveMultipleResponse** element.

3.1.4.10.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.10.2.1 ResolveMultiple

The input data for the **ResolveMultiple** WSDL operation.

```
<xs:element name="ResolveMultiple">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The list of input strings to be resolved. This MUST NOT be NULL. If this is NULL, the protocol server MUST return an **ArgumentNullException<"resolveInput">** message.

The protocol server MUST resolve each string across all claims providers.

3.1.4.10.2.2 ResolveMultipleResponse

The result data for the **ResolveMultiple** WSDL operation.

```
<xs:element name="ResolveMultipleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleResult" 
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResolveMultipleResult: The list of picker entities.

The list MUST contain one and only one picker entity per string in the input.

If there are multiple matches for an input string, then the protocol server MUST return a picker entity whose IsResolved is set to false and contains the multiple matched picker entities as the child entities.

3.1.4.10.3 Complex Types

None.

3.1.4.10.4 Simple Types

None.

3.1.4.10.5 Attributes

None.

3.1.4.10.6 Groups

None.

3.1.4.10.7 Attribute Groups

None.

3.1.4.11 ResolveMultipleClaim

This operation is used to resolve a list of claims to picker entities using a list of claims providers.

```
<wsdl:operation name="ResolveMultipleClaim">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleClaim" message="tns:IClaimProviderWebService_ResolveMultipleClaim_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleClaimResponse"
message="tns:IClaimProviderWebService_ResolveMultipleClaim_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ResolveMultipleClaim_InputMessage** (section 3.1.4.11.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveMultipleClaim_OutputMessage** (section 3.1.4.11.1.2) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.11.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.11.1.1 IClaimProviderWebService_ResolveMultipleClaim_InputMessage

The request WSDL message for the **ResolveMultipleClaim** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaim
```

The SOAP body contains the **ResolveMultipleClaim** element.

3.1.4.11.1.2 IClaimProviderWebService_ResolveMultipleClaim_OutputMessage

The response WSDL message for the **ResolveMultipleClaim** WSDL operation.

The SOAP body contains the **ResolveMultipleClaimResponse** element.

3.1.4.11.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.11.2.1 ResolveMultipleClaim

The input data for the **ResolveMultipleClaim** WSDL operation.

```
<xs:element name="ResolveMultipleClaim">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:ArrayOfSPClaim"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The list of claims (2) to be resolved. This MUST NOT be NULL. If this is NULL, the protocol server MUST return an **ArgumentNullException<"searchPattern">** message.

3.1.4.11.2.2 ResolveMultipleClaimResponse

The result data for the **ResolveMultipleClaim** WSDL operation.

```
<xs:element name="ResolveMultipleClaimResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleClaimResult"
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

ResolveMultipleClaimResult: The list of picker entities. The list MUST contain one and only one picker entity per one claim (2) in the input. If there are multiple matches for an input claim, the protocol server MUST return a picker entity whose **IsResolved** is set to **false** and contains the multiple matched picker entity as the child entities.

3.1.4.11.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.11.3.1 ArrayOfSPClaim

An array of elements of type **SPClaim** (section 2.2.4.10).

```
<xs:complexType name="ArrayOfSPClaim">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPClaim" nillable="true"
type="tns:SPClaim"/>
  </xs:sequence>
</xs:complexType>
```

SPClaim: The claim (2) associated with an entity.

3.1.4.11.4 Simple Types

None.

3.1.4.11.5 Attributes

None.

3.1.4.11.6 Groups

None.

3.1.4.11.7 Attribute Groups

None.

3.1.4.12 Search

This operation is used to perform a search for entities on a list of claims providers.

The input to this operation is a search string and the result of this operation is a list of claims provider hierarchy trees that contain the matched entities. If a hierarchy provider is available, the protocol server MAY move results returned by claims providers to hierarchy provider hierarchy tree.

```
<wsdl:operation name="Search">
  <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search"
message="tns:IClaimProviderWebService_Search_InputMessage"/>
  <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchRe
sponse" message="tns:IClaimProviderWebService_Search_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_Search_InputMessage** (section 3.1.4.12.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_Search_OutputMessage** (section 3.1.4.12.1.2) response message.

The protocol server MUST search across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider search arguments in the input message.
- The claims providers support search.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.12.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.12.1.1 IClaimProviderWebService_Search_InputMessage

The request WSDL message for the **Search** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search
```

The SOAP body contains the **Search** element.

3.1.4.12.1.2 IClaimProviderWebService_Search_OutputMessage

The response WSDL message for the **Search** WSDL operation.

The SOAP body contains the **SearchResponse** element.

3.1.4.12.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.12.2.1 Search

The input data for the **Search** WSDL operation.

```
<xs:element name="Search">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerSearchArguments"
type="tns:ArrayOfSPProviderSearchArguments"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerSearchArguments: The list of **SPProviderSearchArguments** (section 3.1.4.12.3.2) that specify what claims providers to search, where to begin the search in each claims provider, and how many entities can be included in the output.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

searchPattern: The search string. The protocol server MUST search for the string in each claims provider. If this is NULL, the protocol server MUST return an **ArgumentNullException<"searchPattern">** message.

3.1.4.12.2.2 SearchResponse

The result data for the **Search** WSDL operation.

```
<xs:element name="SearchResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="SearchResult"
type="tns:ArrayOfSPProviderHierarchyTree"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SearchResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that contains entities that match the search string.

3.1.4.12.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.12.3.1 ArrayOfSPProviderSearchArguments

An array of **SPProviderSearchArguements** (section 3.1.4.12.3.2**).**

```
<xs:complexType name="ArrayOfSPProviderSearchArguments">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderSearchArguments"
nillable="true" type="tns:SPProviderSearchArguments"/>
  </xs:sequence>
</xs:complexType>
```

SPProviderSearchArguments: The information to control search in a claims provider.

3.1.4.12.3.2 SPProviderSearchArguments

The information to control search in a claims provider.

```
<xs:complexType name="SPProviderSearchArguments">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
```

```
      <xs:element minOccurs="1" maxOccurs="1" name="MaxCount" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
```

ProviderName: The name of a claims provider.

HierarchyNodeID: The node identifier of a node in claims provider hierarchy tree where the protocol server begins the search. If NULL is specified, then the protocol server MUST begin the search at the root of the claims provider hierarchy tree.

MaxCount: The maximum number of matched entities to be returned by the protocol server in the output claims provider hierarchy tree for the specified claims provider. This value MUST be greater than or equal to 1.

3.1.4.12.4 Simple Types

None.

3.1.4.12.5 Attributes

None.

3.1.4.12.6 Groups

None.

3.1.4.12.7 Attribute Groups

None.

3.1.4.13 SearchAll

This operation is used to perform a search for entities on a list of claims providers. The result of this operation is a list of claims provider hierarchy trees. If a hierarchy provider is available, the protocol server MAY move entities returned by claims providers to hierarchy provider hierarchy tree.

```
<wsdl:operation name="SearchAll">
  <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAl
l" message="tns:IClaimProviderWebService_SearchAll_InputMessage"/>
  <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAl
lResponse" message="tns:IClaimProviderWebService_SearchAll_OutputMessage"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_SearchAll_InputMessage** (section 3.1.4.13.1.1) request WSDL message and the protocol server responds with an **IClaimProviderWebService_SearchAll_OutputMessage** (section 3.1.4.13.1.2) response message.

The protocol server MUST search across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (2) specified in the input message.
- The claims providers are listed in the provider names in the input message.

- The claims providers support search.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.13.1 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.13.1.1 IClaimProviderWebService_SearchAll_InputMessage

The request WSDL message for the **SearchAll** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll
```

The SOAP body contains the **SearchAll** element.

3.1.4.13.1.2 IClaimProviderWebService_SearchAll_OutputMessage

The response WSDL message for the **SearchAll** WSDL operation.

The SOAP body contains the **SearchAllResponse** element.

3.1.4.13.2 Elements

The following XML Schema element definitions are specific to this operation.

3.1.4.13.2.1 SearchAll

The input data for the **SearchAll** WSDL operation.

```
<xs:element name="SearchAll">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="maxCount" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is NULL, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

searchPattern: The search string. This MUST NOT be NULL. If list of provider names is NULL, the protocol server MUST search for each string in all claims providers. If this is NULL, the protocol server MUST return an **ArgumentNullException<"searchPattern">** message.

maxCount: The maximum number of matched entities to be returned in total across all the output claims provider hierarchy trees. This number MUST be greater than or equal to 1.

3.1.4.13.2.2 SearchAllResponse

The result data for the **SearchAll** WSDL operation.

```
<xs:element name="SearchAllResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="SearchAllResult" 
type="tns:ArrayOfSPProviderHierarchyTree"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SearchAllResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that contains entities that match the search string.

3.1.4.13.3 Complex Types

None.

3.1.4.13.4 Simple Types

None.

3.1.4.13.5 Attributes

None.

3.1.4.13.6 Groups

None.

3.1.4.13.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

4.1 GetHierarchyAll

The protocol server has four claims providers: System, AllUsers, Group and People. The System **claims provider** has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

4.1.1 Retrieve Claim Provider Hierarchy Trees from All Available Claim Providers

In this example, claims provider hierarchy trees are fetched from all available claims providers. The protocol server has four claims providers: System, AllUsers, Group and People.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchyAll xmlns="http://tempuri.org/"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <numberOfLevels>3</numberOfLevels>
    </GetHierarchyAll>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyAllResponse xmlns="http://tempuri.org/">
      <GetHierarchyAllResult>
        <SPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>

<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
            <SPProviderHierarchyNode>
              <Nm>Farm ID</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
ID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
          </Children>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
        <SPProviderHierarchyTree>
          <Nm>AllUsers</Nm>
          <ProviderName>AllUsers</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>true</IsLeaf>
          <Children/>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
        <SPProviderHierarchyTree>
          <Nm>Group</Nm>
          <ProviderName>Group</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>true</IsLeaf>
          <Children/>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
        <SPProviderHierarchyTree>
          <Nm>People and Organizations</Nm>
          <ProviderName>People</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>true</IsLeaf>
          <Children/>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
      </GetHierarchyAllResult>
    </GetHierarchyAllResponse>
  </s:Body>
</s:Envelope>
```

4.1.2 Retrieve Claim Provider Hierarchy Trees from One of the Claim Providers

In this example, claims provider hierarchy trees are fetched from one of the claims providers ("System").

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchyAll xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
      <numberOfLevels>3</numberOfLevels>
      <includeEntityData>true</includeEntityData>
    </GetHierarchyAll>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyAllResponse xmlns="http://tempuri.org/">
      <GetHierarchyAllResult>
        <SPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
            <SPProviderHierarchyNode>
              <Nm>Farm ID</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
ID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
          </Children>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
      </GetHierarchyAllResult>
    </GetHierarchyAllResponse>
    </s:Body>
</s:Envelope>
```

4.2 SearchAll

In this example, protocol client calls the SearchAll method to search the entities on two claims providers ("System" and "All Users") with a search pattern "domain\user1". The **URI** of the Web application (2) (http://localhost) and the maximum number of matched entities to be returned (100) are also specified.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <SearchAll xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
        <string>Groups</string>
      </providerNames>
      <searchPattern>Domain\User1</searchPattern>
      <maxCount>100</maxCount>
    </SearchAll>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SearchAllResponse xmlns="http://tempuri.org/">
      <SearchAllResult>
        <SPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData>
                <PickerEntity>
                  <Key>c:0'0c|System|Domain\User1</Key>
                  <DisplayText>Domain\User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                  <IsResolved>true</IsResolved>
                  <Description/>
                  <EntityType>System</EntityType>
                  <EntityDataElements/>
                  <MultipleMatches/>
                </PickerEntity>
              </EntityData>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
            <SPProviderHierarchyNode>
              <Nm>Farm ID</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
ID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
          </Children>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
      </SearchAllResult>
    </SearchAllResponse>
    </s:Body>
</s:Envelope>
```

4.3 Search

In this example, protocol client calls the Search method to search the entities on one of the claims providers ("System") with a search pattern "domain\user1". The URI of the Web application (http://localhost) and the maximum number of matched entities to be returned (100) are also specified.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <Search xmlns="http://tempuri.org/"      xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerSearchArguments>
        <SPProviderSearchArguments>
          <ProviderName>System</ProviderName>
          <MaxCount>100</MaxCount>
        </SPProviderSearchArguments>
      </providerSearchArguments>
      <searchPattern> domain\user1</searchPattern>
    </Search>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SearchResponse xmlns="http://tempuri.org/">
      <SearchResult>
        <SPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData>
                <PickerEntity>
                  <Key>c:0'0c|System|Domain\User1</Key>
                  <DisplayText>Domain\User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                  <IsResolved>true</IsResolved>
                  <Description/>
                  <EntityType>System</EntityType>
                  <EntityDataElements/>
                  <MultipleMatches/>
                </PickerEntity>
              </EntityData>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
            <SPProviderHierarchyNode>
              <Nm>Farm ID</Nm>
              <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
ID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPProviderHierarchyNode>
          </Children>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPProviderHierarchyTree>
      </SearchResult>
    </SearchResponse>
    </s:Body>
</s:Envelope>
```

4.4 GetHierarchy

In this example, protocol client retrieves a claims provider hierarchy tree from a claims provider by specifying node identifier of the claims provider hierarchy tree.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchy xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerName>System</providerName>

<hierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</hierarchyNodeID>
      <numberOfLevels>4</numberOfLevels>
      <includeEntityData>true</includeEntityData>
    </GetHierarchy>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyResponse xmlns="http://tempuri.org/">
      <GetHierarchyResult>
        <Nm>Application Pool Identity</Nm>
        <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
ame</HierarchyNodeID>
        <IsLeaf>true</IsLeaf>
        <Children/>
        <EntityData/>
        <Count>-1</Count>
        <IsRoot>false</IsRoot>
      </GetHierarchyResult>
    </GetHierarchyResponse>
    </s:Body>
</s:Envelope>
```

4.5 EntityTypes

In this example, protocol client fetches the picker entity types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
```

```
    <EntityTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </EntityTypes>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <EntityTypesResponse xmlns="http://tempuri.org/">
      <EntityTypesResult>
        <string>System</string>
      </EntityTypesResult>
    </EntityTypesResponse>
    </s:Body>
</s:Envelope>
```

4.6 ClaimTypes

In this example, protocol client fetches the claim types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ClaimTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </ClaimTypes>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ClaimTypesResponse xmlns="http://tempuri.org/">
<ClaimTypesResult>
<string>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</string>
<string>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</stri
```

```
ng>
<string>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitysid</string>

<string>http://schemas.microsoft.com/sharepoint/2009/08/claims/windowstoken/handle</string>
<string>http://sharepoint.microsoft.com/claims/2009/01/windowstoken/processid</string>
      </ClaimTypesResult>
    </ClaimTypesResponse>
    </s:Body>
</s:Envelope>
```

4.7 ClaimValueTypes

In this example, protocol client fetches the claim value types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (2) (http://localhost) that is configured to use claims authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ClaimValueTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </ClaimValueTypes>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ClaimValueTypesResponse xmlns="http://tempuri.org/">
      <ClaimValueTypesResult>
        <string>http://www.w3.org/2001/XMLSchema#Guid</string>
        <string>http://www.w3.org/2001/XMLSchema#string</string>
      </ClaimValueTypesResult>
    </ClaimValueTypesResponse>
    </s:Body>
 </s:Envelope>
```

4.8 Resolve

In this example, protocol client calls **Resolve** method to resolve an input string ("domain\user1") to picker entities using the available claims providers.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types: Application Pool Identity and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <Resolve xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <resolveInput>domain\user1</resolveInput>
    </Resolve>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ResolveResponse xmlns="http://tempuri.org/">
      <ResolveResult>
        <PickerEntity>
          <Key>c:0'0c|System| domain\user1</Key>
          <DisplayText>user1</DisplayText>
          <IsResolved>true</IsResolved>
          <Description/>
          <EntityType>System</EntityType>
          <EntityDataElements/>
          <MultipleMatches/>
        </PickerEntity>
        <PickerEntity>
          <Key>i:0#0w| domain\user1</Key>
          <DisplayText>User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
          <IsResolved>true</IsResolved>
          <Description> domain\user1</Description>
          <EntityType>User</EntityType>
          <EntityGroupName>User</EntityGroupName>
          <EntityDataElements>
            <Pair>
              <First xsi:type="xsd:string">Email</First>
              <Second xsi:type="xsd:string"/>
            </Pair>
            <Pair>
              <First xsi:type="xsd:string">SIPAddress</First>
            </Pair>
            <Pair>
              <First xsi:type="xsd:string">Title</First>
              <Second xsi:type="xsd:string"/>
            </Pair>
            <Pair>
```

```
              <First xsi:type="xsd:string">WorkPhone</First>
              <Second xsi:type="xsd:string"/>
            </Pair>
          </EntityDataElements>
          <MultipleMatches/>
        </PickerEntity>
      </ResolveResult>
    </ResolveResponse>
    </s:Body>
</s:Envelope>
```

4.9 ResolveMultiple

In this example, protocol client calls **ResolveMulitiple** method to resolve two input strings ("domain\user1" and "domain\user2") to picker entities using the available claims providers.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has the following claim types: Application Pool Identity, and Farm identifier. The protocol server has a Web application (2) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ResolveMultiple xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <resolveInput>
        <string>domain\user1</string>
        <string>domain\user2</string>
      </resolveInput>
    </ResolveMultiple>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ResolveMultipleResponse xmlns="http://tempuri.org/">
      <ResolveMultipleResult>
        <PickerEntity>
          <Key/>
          <DisplayText>domain\user1</DisplayText>
          <IsResolved>false</IsResolved>
          <Description/>
          <EntityType/>
          <EntityDataElements/>
```

```
            <MultipleMatches>
              <anyType xsi:type="PickerEntity">
                <Key>c:0'0c|System|domain\user1</Key>
                <DisplayText>User1
  (/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                <IsResolved>true</IsResolved>
                <Description/>
                <EntityType>System</EntityType>
                <EntityDataElements/>
                <MultipleMatches/>
              </anyType>
              <anyType xsi:type="PickerEntity">
                <Key>i:0#0w|domain\user1</Key>
                <DisplayText>User1</DisplayText>
                <IsResolved>true</IsResolved>
                <Description>domain\user1</Description>
                <EntityType>User</EntityType>
                <EntityDataElements>
                  <Pair>
                    <First xsi:type="xsd:string">Email</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">SIPAddress</First>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">Title</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">WorkPhone</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                </EntityDataElements>
                <MultipleMatches/>
              </anyType>
            </MultipleMatches>
          </PickerEntity>
          <PickerEntity>
            <Key/>
            <DisplayText>domain\user2</DisplayText>
            <IsResolved>false</IsResolved>
            <Description/>
            <EntityType/>
            <EntityDataElements/>
            <MultipleMatches>
              <anyType xsi:type="PickerEntity">
                <Key>c:0'0c|System|domain\user2</Key>
                <DisplayText>User2
  (/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                <IsResolved>true</IsResolved>
                <Description/>
                <EntityType>System</EntityType>
                <EntityDataElements/>
                <MultipleMatches/>
              </anyType>
              <anyType xsi:type="PickerEntity">
                <Key>i:0#0w|domain\user2</Key>
                <DisplayText>User2</DisplayText>
                <IsResolved>true</IsResolved>
                <Description>domain\user2</Description>
                <EntityType>User</EntityType>
                <EntityDataElements>
                  <Pair>
                    <First xsi:type="xsd:string">Email</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">SIPAddress</First>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">Title</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                  <Pair>
                    <First xsi:type="xsd:string">WorkPhone</First>
                    <Second xsi:type="xsd:string"/>
                  </Pair>
                </EntityDataElements>
                <MultipleMatches/>
              </anyType>
            </MultipleMatches>
          </PickerEntity>
        </ResolveMultipleResult>
      </ResolveMultipleResponse>
      </s:Body>
  </s:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

This protocol introduces no additional security considerations beyond those applicable to its underlying protocols.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided below:

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:i0="http://schemas.microsoft.com/sharepoint/claims/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.microsoft.com/sharepoint/claims/" name="SPClaimProviderWebService"
targetNamespace="http://schemas.microsoft.com/sharepoint/claims/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/sharepoint/claims/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="ClaimTypes">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfString">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:element name="ClaimTypesResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ClaimTypesResult"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ClaimValueTypes">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ClaimValueTypesResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ClaimValueTypesResult"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="EntityTypes">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="EntityTypesResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="EntityTypesResult"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ProviderSchemas">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ProviderSchemasResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchemasResult"
type="tns:ArrayOfSPProviderSchema" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfSPProviderSchema">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderSchema"
nillable="true" type="tns:SPProviderSchema" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="SPProviderSchema">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchema"
type="tns:ArrayOfSPSchemaElement" />
          <xs:element minOccurs="1" maxOccurs="1" name="SupportsHierarchy" type="xs:boolean"
/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfSPSchemaElement">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPSchemaElement"
nillable="true" type="tns:SPSchemaElement" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="SPSchemaElement">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SPSchemaElementType"
/>
        </xs:sequence>
      </xs:complexType>
      <xs:simpleType name="SPSchemaElementType">
        <xs:restriction base="xs:string">
          <xs:enumeration value="None" />
          <xs:enumeration value="TableViewOnly" />
          <xs:enumeration value="DetailViewOnly" />
          <xs:enumeration value="Both" />
        </xs:restriction>
      </xs:simpleType>
      <xs:element name="HierarchyProviderSchema">
        <xs:complexType />
      </xs:element>
      <xs:element name="HierarchyProviderSchemaResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="HierarchyProviderSchemaResult" 
type="tns:SPProviderSchema" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="GetHierarchyAll">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames" 
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:simpleType name="SPPrincipalType">
        <xs:list>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="None" />
              <xs:enumeration value="User" />
              <xs:enumeration value="DistributionList" />
              <xs:enumeration value="SecurityGroup" />
              <xs:enumeration value="SharePointGroup" />
              <xs:enumeration value="All" />
            </xs:restriction>
          </xs:simpleType>
        </xs:list>
      </xs:simpleType>
      <xs:element name="GetHierarchyAllResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyAllResult" 
type="tns:ArrayOfSPProviderHierarchyTree" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfSPProviderHierarchyTree">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderHierarchyTree" 
nillable="true" type="tns:SPProviderHierarchyTree" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="SPProviderHierarchyTree">
        <xs:complexContent mixed="false">
          <xs:extension base="tns:SPProviderHierarchyElement">
            <xs:sequence>
              <xs:element minOccurs="1" maxOccurs="1" name="IsRoot" type="xs:boolean" />
            </xs:sequence>
          </xs:extension>
        </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="SPProviderHierarchyElement">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="Nm" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="IsLeaf" type="xs:boolean" />
          <xs:element minOccurs="0" maxOccurs="1" name="Children"
type="tns:ArrayOfSPProviderHierarchyNode" />
          <xs:element minOccurs="0" maxOccurs="1" name="EntityData"
type="tns:ArrayOfPickerEntity" />
          <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfSPProviderHierarchyNode">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderHierarchyNode"
nillable="true" type="tns:SPProviderHierarchyNode" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="SPProviderHierarchyNode">
        <xs:complexContent mixed="false">
          <xs:extension base="tns:SPProviderHierarchyElement" />
        </xs:complexContent>
      </xs:complexType>
      <xs:complexType name="ArrayOfPickerEntity">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="PickerEntity" nillable="true"
type="tns:PickerEntity" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="PickerEntity">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="Key" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="DisplayText" type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="IsResolved" type="xs:boolean" />
          <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="EntityType" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="EntityGroupName" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="HierarchyIdentifier" />
          <xs:element minOccurs="0" maxOccurs="1" name="EntityDataElements"
type="tns:ArrayOfPair" />
          <xs:element minOccurs="0" maxOccurs="1" name="MultipleMatches"
type="tns:ArrayOfAnyType" />
          <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfPair">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="Pair" nillable="true"
type="tns:Pair" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="Pair">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="First" />
          <xs:element minOccurs="0" maxOccurs="1" name="Second" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfAnyType">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true" />
        </xs:sequence>
      </xs:complexType>
      <xs:element name="GetHierarchy">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerName" type="xs:string" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="hierarchyNodeID" type="xs:string" 
/>
            <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="GetHierarchyResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyResult" 
type="tns:SPProviderHierarchyTree" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="SearchAll">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames" 
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string" />
            <xs:element minOccurs="1" maxOccurs="1" name="maxCount" type="xs:int" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="SearchAllResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="SearchAllResult" 
type="tns:ArrayOfSPProviderHierarchyTree" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Search">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerSearchArguments" 
type="tns:ArrayOfSPProviderSearchArguments" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfSPProviderSearchArguments">
```

```
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderSearchArguments" 
nillable="true" type="tns:SPProviderSearchArguments" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="SPProviderSearchArguments">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="MaxCount" type="xs:int" />
        </xs:sequence>
      </xs:complexType>
      <xs:element name="SearchResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="SearchResult" 
type="tns:ArrayOfSPProviderHierarchyTree" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Resolve">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames" 
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ResolveResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ResolveResult" 
type="tns:ArrayOfPickerEntity" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ResolveClaim">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames" 
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType" 
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:SPClaim" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="SPClaim">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="ClaimType" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="ValueType" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="OriginalIssuer" type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:element name="ResolveClaimResponse">
```

```
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ResolveClaimResult"
type="tns:ArrayOfPickerEntity" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ResolveMultiple">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="resolveInput"
type="tns:ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ResolveMultipleResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleResult"
type="tns:ArrayOfPickerEntity" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ResolveMultipleClaim">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString" />
            <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType" />
            <xs:element minOccurs="0" maxOccurs="1" name="resolveInput"
type="tns:ArrayOfSPClaim" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfSPClaim">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="SPClaim" nillable="true"
type="tns:SPClaim" />
        </xs:sequence>
      </xs:complexType>
      <xs:element name="ResolveMultipleClaimResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleClaimResult"
type="tns:ArrayOfPickerEntity" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="IClaimProviderWebService_ClaimTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:ClaimTypes" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ClaimTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:ClaimTypesResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ClaimValueTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:ClaimValueTypes" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ClaimValueTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:ClaimValueTypesResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_EntityTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:EntityTypes" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_EntityTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:EntityTypesResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ProviderSchemas_InputMessage">
    <wsdl:part name="parameters" element="tns:ProviderSchemas" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ProviderSchemas_OutputMessage">
    <wsdl:part name="parameters" element="tns:ProviderSchemasResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_HierarchyProviderSchema_InputMessage">
    <wsdl:part name="parameters" element="tns:HierarchyProviderSchema" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_HierarchyProviderSchema_OutputMessage">
    <wsdl:part name="parameters" element="tns:HierarchyProviderSchemaResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_GetHierarchyAll_InputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyAll" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_GetHierarchyAll_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyAllResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_GetHierarchy_InputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchy" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_GetHierarchy_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_SearchAll_InputMessage">
    <wsdl:part name="parameters" element="tns:SearchAll" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_SearchAll_OutputMessage">
    <wsdl:part name="parameters" element="tns:SearchAllResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_Search_InputMessage">
    <wsdl:part name="parameters" element="tns:Search" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_Search_OutputMessage">
    <wsdl:part name="parameters" element="tns:SearchResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_Resolve_InputMessage">
    <wsdl:part name="parameters" element="tns:Resolve" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_Resolve_OutputMessage">
    <wsdl:part name="parameters" element="tns:ResolveResponse" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ResolveClaim_InputMessage">
    <wsdl:part name="parameters" element="tns:ResolveClaim" />
  </wsdl:message>
  <wsdl:message name="IClaimProviderWebService_ResolveClaim_OutputMessage">
```

```
      <wsdl:part name="parameters" element="tns:ResolveClaimResponse" />
    </wsdl:message>
    <wsdl:message name="IClaimProviderWebService_ResolveMultiple_InputMessage">
      <wsdl:part name="parameters" element="tns:ResolveMultiple" />
    </wsdl:message>
    <wsdl:message name="IClaimProviderWebService_ResolveMultiple_OutputMessage">
      <wsdl:part name="parameters" element="tns:ResolveMultipleResponse" />
    </wsdl:message>
    <wsdl:message name="IClaimProviderWebService_ResolveMultipleClaim_InputMessage">
      <wsdl:part name="parameters" element="tns:ResolveMultipleClaim" />
    </wsdl:message>
    <wsdl:message name="IClaimProviderWebService_ResolveMultipleClaim_OutputMessage">
      <wsdl:part name="parameters" element="tns:ResolveMultipleClaimResponse" />
    </wsdl:message>
    <wsdl:portType name="IClaimProviderWebService">
      <wsdl:operation name="ClaimTypes">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTyp
es" message="tns:IClaimProviderWebService_ClaimTypes_InputMessage" />
        <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTyp
esResponse" message="tns:IClaimProviderWebService_ClaimTypes_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="ClaimValueTypes">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimVal
ueTypes" message="tns:IClaimProviderWebService_ClaimValueTypes_InputMessage" />
        <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimVal
ueTypesResponse" message="tns:IClaimProviderWebService_ClaimValueTypes_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="EntityTypes">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTy
pes" message="tns:IClaimProviderWebService_EntityTypes_InputMessage" />
        <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTy
pesResponse" message="tns:IClaimProviderWebService_EntityTypes_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="ProviderSchemas">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Provider
Schemas" message="tns:IClaimProviderWebService_ProviderSchemas_InputMessage" />
        <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Provider
SchemasResponse" message="tns:IClaimProviderWebService_ProviderSchemas_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="HierarchyProviderSchema">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarch
yProviderSchema" message="tns:IClaimProviderWebService_HierarchyProviderSchema_InputMessage" 
/>
        <wsdl:output 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarch
yProviderSchemaResponse" 
message="tns:IClaimProviderWebService_HierarchyProviderSchema_OutputMessage" />
      </wsdl:operation>
      <wsdl:operation name="GetHierarchyAll">
        <wsdl:input 
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyAll" message="tns:IClaimProviderWebService_GetHierarchyAll_InputMessage" />
```

```
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyAllResponse" message="tns:IClaimProviderWebService_GetHierarchyAll_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="GetHierarchy">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchy" message="tns:IClaimProviderWebService_GetHierarchy_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyResponse" message="tns:IClaimProviderWebService_GetHierarchy_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="SearchAll">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAl
l" message="tns:IClaimProviderWebService_SearchAll_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAl
lResponse" message="tns:IClaimProviderWebService_SearchAll_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="Search">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search"
message="tns:IClaimProviderWebService_Search_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchRe
sponse" message="tns:IClaimProviderWebService_Search_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="Resolve">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve"
message="tns:IClaimProviderWebService_Resolve_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveR
esponse" message="tns:IClaimProviderWebService_Resolve_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="ResolveClaim">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laim" message="tns:IClaimProviderWebService_ResolveClaim_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laimResponse" message="tns:IClaimProviderWebService_ResolveClaim_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="ResolveMultiple">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultiple" message="tns:IClaimProviderWebService_ResolveMultiple_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleResponse" message="tns:IClaimProviderWebService_ResolveMultiple_OutputMessage" />
    </wsdl:operation>
    <wsdl:operation name="ResolveMultipleClaim">
      <wsdl:input
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleClaim" message="tns:IClaimProviderWebService_ResolveMultipleClaim_InputMessage" />
      <wsdl:output
wsam:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveM
ultipleClaimResponse"
message="tns:IClaimProviderWebService_ResolveMultipleClaim_OutputMessage" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="CustomBinding_IClaimProviderWebService"
type="tns:IClaimProviderWebService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="ClaimTypes">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimType
s" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ClaimValueTypes">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValu
eTypes" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EntityTypes">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTyp
es" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ProviderSchemas">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderS
chemas" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="HierarchyProviderSchema">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarchy
ProviderSchema" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetHierarchyAll">
```

```
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierar
chyAll" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetHierarchy">
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierar
chy" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SearchAll">
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll
" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Search">
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search" 
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Resolve">
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve" 
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ResolveClaim">
      <soap:operation 
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveCl
aim" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ResolveMultiple">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMu
ltiple" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ResolveMultipleClaim">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMu
ltipleClaim" style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Lync™ 2010
- Microsoft® FAST™ Search Server 2010
- Microsoft® Office 2010 suites
- Microsoft® SharePoint® Designer 2010
- Microsoft® SharePoint® Foundation 2010
- Microsoft® SharePoint® Workspace 2010
- Microsoft® Visio® 2010

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

8 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

9 Index