# [MS-WCFESAN-Diff]:

# WCF-Based Encrypted Server Administration and Notification Protocol

**Intellectual Property Rights Notice for Open Specifications Documentation**

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards as well as overviews of the interaction among each of these technologiessupport. Additionally, overview documents cover inter-protocol relationships and interactions.

- **Copyrights**. This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you maycan make copies of it in order to develop implementations of the technologies that are described in the Open Specifications this documentation and maycan distribute portions of it in your implementations usingthat use these technologies or in your documentation as necessary to properly document the implementation. You maycan also distribute in your implementation, with or without modification, any schema, IDL'sschemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications. documentation.

- **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.

- **Patents**. Microsoft has patents that maymight cover your implementations of the technologies described in the Open Specifications. documentation. Neither this notice nor Microsoft's delivery of thethis documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specification maySpecifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in the Open Specificationsthis documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **Trademarks**. The names of companies and products contained in this documentation maymight be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.

- **Fictitious Names**. The example companies, organizations, products, domain names, e-mailemail addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights**. All other rights are reserved, and this notice does not grant any rights other than as specifically described above, whether by implication, estoppel, or otherwise.

**Tools**. The Open Specifications dodocumentation does not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments, you are free to take advantage of them. Certain Open Specifications documents are intended for use in conjunction with publicly available standardstandards specifications and network programming art, and assumes, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

## Revision Summary

| Date | Revision History | Revision Class | Comments |
|---|---|---|---|
| 7/12/2012 | 1.0 | New | Released new document. |
| 10/25/2012 | 2.0 | Major | Significantly changed the technical content. |
| 1/31/2013 | 3.0 | Major | Significantly changed the technical content. |
| 8/8/2013 | 4.0 | Major | Significantly changed the technical content. |
| 11/14/2013 | 5.0 | Major | Significantly changed the technical content. |
| 2/13/2014 | 5.0 | None | No change to the meaning, language, or formatting of the technical content. |
| 5/15/2014 | 5.0 | None | No change to the meaning, language, or formatting of the technical content. |
| 6/30/2015 | 6.0 | Major | Significantly changed the technical content. |
| 10/16/2015 | 6.0 | No ChangeNone | No changes to the meaning, language, or formatting of the technical content. |

# Table of Contents

*[MS-WCFESAN-Diff] - v20160714*
*WCF-Based Encrypted Server Administration and Notification Protocol*
*Copyright © 2016 Microsoft Corporation*
*Release: July 14, 2016*

*[MS-WCFESAN-Diff] - v20160714*
*WCF-Based Encrypted Server Administration and Notification Protocol*
*Copyright © 2016 Microsoft Corporation*
*Release: July 14, 2016*

*[MS-WCFESAN-Diff] - v20160714*
*WCF-Based Encrypted Server Administration and Notification Protocol*
*Copyright © 2016 Microsoft Corporation*
*Release: July 14, 2016*

# 1 Introduction

The WCF-Based Encrypted Server Administration and Notification Protocol (Provider Framework) enables the protocol client to monitor and manage the protocol server in the same network.

This protocol is an extension to the .NET Message Framing TCP Binding Protocol [MS-NMFTB].

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [RFC2119]. Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

## 1.1 Glossary

TheThis document uses the following terms are specific to this document:

**Active Directory**: A general-purpose network directory service. **Active Directory** also refers to the Windows implementation of a directory service. **Active Directory** stores information about a variety of objects in the network. Importantly, user accounts, computer accounts, groups, and all related credential information used by the Windows implementation of Kerberos are stored in **Active Directory**. **Active Directory** is either deployed as Active Directory Domain Services (AD DS) or Active Directory Lightweight Directory Services (AD LDS). [MS-ADTS] describes both forms. For more information, see [MS-AUTHSOD] section 1.1.1.5.2, Lightweight Directory Access Protocol (LDAP) versions 2 and 3, Kerberos, and **DNS**.

**Coordinated Universal Time (UTC)**: A high-precision atomic time standard that approximately tracks Universal Time (UT). It is the basis for legal, civil time all over the Earth. Time zones around the world are expressed as positive and negative offsets from UTC. In this role, it is also referred to as Zulu time (Z) and Greenwich Mean Time (GMT). In these specifications, all references to UTC refer to the time at UTC-0 (or GMT).

**Domain Name System (DNS)**: A hierarchical, distributed database that contains mappings of domain names (1) to various types of data, such as IP addresses. DNS enables the location of computers and services by user-friendly names, and it also enables the discovery of other information stored in the database.

**fully qualified domain name (FQDN)**: An unambiguous domain name (2) that gives an absolute location in the **Domain Name System's (DNS)** hierarchy tree, as defined in [RFC1035] section 3.1 and [RFC2181] section 11.

**globally unique identifier (GUID)**: A term used interchangeably with universally unique identifier (UUID) in Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the value. Specifically, the use of this term does not imply or require that the algorithms described in [RFC4122] or [C706] must be used for generating the **GUID**. See also universally unique identifier (UUID).

**security identifier (SID)**: An identifier for security principals in Windows that is used to identify an account or a group. Conceptually, the **SID** is composed of an account authority portion (typically a domain) and a smaller integer representing an identity relative to the account authority, termed the relative identifier (RID). The **SID** format is specified in [MS-DTYP] section 2.4.2; a string representation of **SIDs** is specified in [MS-DTYP] section 2.4.2 and [MS-AZOD] section 1.1.1.2.

**service (SRV) resource record**: A **Domain Name System (DNS)** resource record used to identify computers that host specific services, as specified in [RFC2782]. SRV resource records are used to locate domain controllers (DCs) for **Active Directory**.

**SOAP**: A lightweight protocol for exchanging structured information in a decentralized, distributed environment. **SOAP** uses XML technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation-specific semantics. SOAP 1.2 supersedes SOAP 1.1. See [SOAP1.2-1/2003].

**SOAP message**: An XML document consisting of a mandatory SOAP envelope, an optional SOAP header, and a mandatory SOAP body. See [SOAP1.2-1/2007] section 5 for more information.

**Transmission Control Protocol (TCP)**: A protocol used with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. TCP handles keeping track of the individual units of data (called packets) that a message is divided into for efficient routing through the Internet.

**Uniform Resource Identifier (URI)**: A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [RFC3986].

**XML namespace**: A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [RFC3986]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [XMLNS-2ED].

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

## 1.2   References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

### 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.

[IANAPORT] IANA, "Service Name and Transport Protocol Port Number Registry", November 2006, http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

[MS-KILE] Microsoft Corporation, "Kerberos Protocol Extensions".

[MS-NMFTB] Microsoft Corporation, ".NET Message Framing TCP Binding Protocol".

[MS-WSPOL] Microsoft Corporation, "Web Services: Policy Assertions and WSDL Extensions".

[RFC1518] Rekhter, Y., Watson, T.J., and Li, T., "An Architecture for IP Address Allocation with CIDR", RFC 1518, September 1993, http://www.ietf.org/rfc/rfc1518.txt

[RFC1519] Fuller, V., Li, T., Yu, J., and Varadhan, K., "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy", RFC 1519, September 1993, https://www.ietf.org/rfc/rfc1519.txt

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

[RFC2246] Dierks, T., and Allen, C., "The TLS Protocol Version 1.0", RFC 2246, January 1999, http://www.rfc-editor.org/rfc/rfc2246.txt

[RFC4120] Neuman, C., Yu, T., Hartman, S., and Raeburn, K., "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005, http://www.rfc-editor.org/rfc/rfc4120.txt

[SOAP1.2-1/2007] Gudgin, M., Hadley, M., Mendelsohn, N., et al., "SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)", W3C Recommendation 27, April 2007, http://www.w3.org/TR/2007/REC-soap12-part1-20070427/

[SOAP1.2-2/2007] Gudgin, M., Hadley, M., Mendelsohn, N., et al., "SOAP Version 1.2 Part 2: Adjuncts (Second Edition)", W3C Recommendation, April 2007, http://www.w3.org/TR/2007/REC-soap12-part2-20070427

[WSADDSoapBind1.0] Gudgin, M., Hadley, M., and Rogers, T., Eds., "Web Services Addressing 1.0 - SOAP Binding", W3C Recommendation, May 2006, http://www.w3.org/TR/2006/REC-ws-addr-soap-20060509/ws-addr-soap.pdf

[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-2ED] World Wide Web Consortium, "Namespaces in XML 1.0 (Second Edition)", August 2006, http://www.w3.org/TR/2006/REC-xml-names-20060816/

[XMLSCHEMA1] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "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

[MSDN-GetLocMachCert] Microsoft Corporation, "MachineIdentityManager.GetLocalMachineCert method", http://msdn.microsoft.com/library/microsoft.windowsserversolutions.devices.identity.machineidentitymanager.getlocalmachinecert.aspx

[MSDN-NETTcpBE] Microsoft Corporation, "TcpTransportBindingElement Class", http://msdn.microsoft.com/en-us/library/system.servicemodel.channels.tcptransportbindingelement.aspx

[MSDN-ProviderException] Microsoft Corporation, "ProviderException Class", http://msdn.microsoft.com/library/microsoft.windowsserversolutions.common.providerframework.providerexception

[MSDN-WSSSDK] Microsoft Corporation, "Windows Server Solutions SDK", http://msdn.microsoft.com/library/gg513877.aspx

[MSKB-2975719] Microsoft Corporation, "August 2014 update rollup for Windows RT 8.1, Windows 8.1, and Windows Server 2012 R2", http://support.microsoft.com/kb/2975719

## 1.3  Overview

The WCF-Based Encrypted Server Administration and Notification Protocol (Provider Framework) defines a series of WCF services that provide basic monitoring and manageability functionalities of the protocol server. Network administrators can use the protocol to monitor and manage the protocol server. It is a classic duplex mode client-server protocol.

The WCF services defined by Provider Framework are described in section 3. The data contracts used by these services are described in section 2.2.

## 1.4 Relationship to Other Protocols

The Provider Framework protocol uses **SOAP messages** to format the requests and responses as specified in [SOAP1.2-1/2007] and [SOAP1.2-2/2007]. These SOAP messages are encrypted by the TcpTransportBindingElement Class described in [MSDN-NETTcpBE] and sent over the .NET Message Framing TCP Binding Protocol [MS-NMFTB], which is built on top of the TCP/IP protocol.
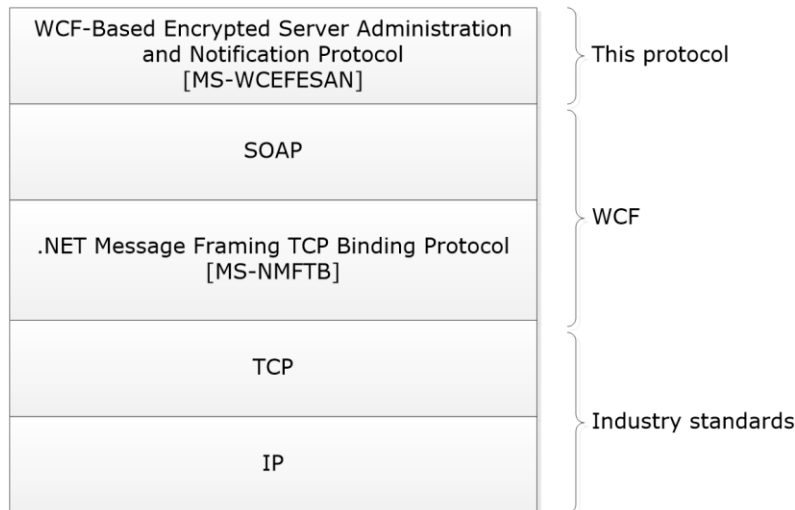


**Figure 1: Protocol stack diagram**

## 1.5 Prerequisites/Preconditions

This protocol operates against a protocol server that exposes a set of WCF service methods that are known by protocol clients.

Provider Framework services needs to be deployed on the protocol server for this protocol to work correctly.

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

## 1.6 Applicability Statement

This protocol is intended for use by protocol clients and protocol servers that are connected by high-bandwidth, low-latency network connections.

## 1.7 Versioning and Capability Negotiation

▪ This document covers versioning issues in the following areas:

▪ **Supported transports:** This protocol can be implemented on top of **TCP** as discussed in section 2.1.

▪ **Protocol versions:** This protocol is not versioned.

▪ **Capability negotiation:** This protocol does not support version negotiation.

## 1.8 Vendor-Extensible Fields

Vendor-extensible fields can extend the Provider Framework by writing their own WCF services. This is done by using the Provider Framework API, which is included in the Windows Server Solutions SDK [MSDN-WSSSDK], to create a provider.

## 1.9 Standards Assignments

The Provider Framework protocol uses the following assignment:

| Parameter | TCP Port | Reference |
|---|---|---|
| wsscomfrmwk | 6602(0x19CA) | [IANAPORT] |

# 2 Messages

## 2.1 Transport

This protocol transports messages using a TCP stream and specifies a framing for messages over a TCP stream. The protocol does not define a mechanism to establish the TCP connection; rather, an established TCP connection is a precondition for this protocol.

This protocol does not define any means for activating a protocol server or protocol client. The protocol server MUST be configured and begin listening in an implementation-specific way. Additionally, the protocol client MUST know the format and transport that is used by the protocol server, for example, the SOAP format over a TCP transport.

The protocol server MUST support formatting of **SOAP** over TCP/IP as specified in [MS-NMFTB].

All protocol messages MUST be transported by using TCP bindings at the transport level.

All protocol messages MUST be formatted as specified in [SOAP1.2-1/2007] or [SOAP1.2-2/2007].

The protocol MUST use Message Framing Security Provider WindowsTransportSecurity as specified in [MS-WSPOL] section 2.2.3.8 with the ProtectionLevel being EncryptAndSign.

## 2.2 Message Syntax

This section contains common definitions used by this protocol. The syntax of the definitions uses XML schema (XSD) as defined in [XMLSCHEMA1] and [XMLSCHEMA2], and Web Services Description Language as defined in [WSDL].

In the following sections, the schema definition might be less restrictive than the processing rules imposed by the protocol. The WSDL in this specification 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.

### 2.2.1 Namespaces

This specification defines and references various **XML namespaces** using the mechanisms specified in [XMLNS-2ED]. Although this specification associates a specific 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 |
|---|---|---|
| None | http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010 | |
| xsd | http://www.w3.org/2001/XMLSchema | [XMLSCHEMA1] [XMLSCHEMA2] |
| soap | http://schemas.xmlsoap.org/wsdl/soap/ | [SOAP1.2-1/2007] [SOAP1.2-2/2007] |
| s | http://www.w3.org/2003/05/soap-envelope | [SOAP1.2-1/2007] |

| Prefix | Namespace URI | Reference |
|---|---|---|
| a | http://www.w3.org/2005/08/addressing | [WSADDSoapBind1.0] |

## 2.2.2 Messages

| Message | Description |
|---|---|
| Common Soap Fault | The structure of the common SOAP fault that is returned by the protocol server to the protocol client when there is an error processing the request. |
| Keepalive | The structure of the Keepalive message sent by the management client to the management server to detect loss of the underlying transport connection. |

### 2.2.2.1 Common SOAP Fault

The SOAP fault message is defined in [SOAP1.2-1/2007]. The following is the specific format used by this protocol.

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Body>
    <s:Fault>
      <s:Code>
        <s:Value>s:Sender</s:Value>
      </s:Code>
      <s:Reason>
        <s:Text xml:lang="en-US">The creator of this fault did not specify a Reason.</s:Text>
      </s:Reason>
      <s:Detail>
        [ProviderException]
      </s:Detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
```

The [ProviderException] element is the part of the message that provides more information about the fault itself. This will be of type Microsoft.WindowsServerSolutions.Common.ProviderFramework.ProviderException [MSDN-ProviderException] or the types that extend the same.

### 2.2.2.2 Keepalive

Following is the Keepalive message that is transmitted by the protocol client in a period of inactivity and ignored by the server. This helps to detect the drop of the TCP session because of inactivity.

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:Action
s:mustUnderstand="1">Microsoft.WindowsServerSolutions.Common.ProviderFramework.Keepalive</a:Action>
  </s:Header>
  <s:Body/>
</s:Envelope>
```

## 2.2.3 Elements

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

## 2.2.4 Complex Types

~~Note: Some of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it in the Product Behavior appendix.~~

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 |
|---|---|
| ClientFileBackupException (section 2.2.4.1.1) | The errors in a client file backup provider. |
| ConfigurationSet (section 2.2.4.1.2) | The configuration settings for client file backups. |
| AutoExclusionSizesInfo (section 2.2.4.2.1) | Information about sizes of data automatically excluded from a backup. |
| PCBackupClientFault (section 2.2.4.2.2) | Information about an error occurring in the computer backup provider service. |
| PCBackupConfiguration (section 2.2.4.2.3) | The entire computer backup configuration as seen from management UI. |
| PCBackupServerFault (section 2.2.4.2.4) | Information about an error occurring in the computer backup server provider service. |
| PCBackupStatus (section 2.2.4.2.5) | The state of the current backup set operation. |
| BadFileInfo (section 2.2.4.2.6) | One file in the backup that contains a bad block. |
| ClientConfig (section 2.2.4.2.7) | All of the client configuration of a backup. |
| ClientConfigurationInfo (section 2.2.4.2.8) | The type of client configuration of a backup. |
| PCClientDataStatus | The status of backups for each client after repair. |
| ClientJobInfo (section 2.2.4.2.9) | Information about one machine-wide backup job. |
| ClientRepairStatus (section 2.2.4.2.10) | The status of backups for each client after repair. |
| ClientVolumeInfo (section 2.2.4.2.11) | Information for a one-client volume. |
| ConfigureBackupFault (section 2.2.4.2.12) | Information about an error occurring when configure backup for a client fails due to another configure backup being in progress. |
| ExcludedFolderInfo (section 2.2.4.2.13) | Information about one excluded folder. |
| FolderInfo (section 2.2.4.2.14) | Information about one folder. |
| FolderInfoGroup (section 2.2.4.2.15) | Information about a group of folders. |
| FolderRequest (section 2.2.4.2.16) | Data needed for Get Folder Children. |
| OperationStatusResponse (section 2.2.4.2.17) | The response of a Get Operation Status request. |

| Complex Type | Description |
|---|---|
| ServiceStatusResponse (section 2.2.4.2.18) | The response of service status request. |
| VolumeJobInfo (section 2.2.4.2.19) | Information about a volume job. |
| WaitingOperation (section 2.2.4.2.20) | Information about a waiting operation. |
| RepairResult (section 2.2.4.2.21) | Information about the result of a repair job. |
| AlertListChunk (section 2.2.4.3.1) | A portion of a list of alerts. |
| AlertInfo (section 2.2.4.3.2) | Information about a Network Health Alert. |
| UserContextTask (section 2.2.4.3.3) | Defines a task that is expected to be launched from a user context. |
| ClearCommand (section 2.2.4.3.4) | Clears an alert. |
| RaiseCommand (section 2.2.4.3.5) | Raises an alert. |
| SuppressCommand (section 2.2.4.3.6) | Suppresses an alert. |
| ReturnClass Type (section 2.2.4.4.1) | Describes the return class for the machine identity provider. |
| GetMachineStatusType (section 2.2.4.4.2) | The return of the get **security identifier (SID)** request. |
| PersonalInfo (section 2.2.4.5.1) | User personal information. |
| UserInfo (section 2.2.4.5.2) | Present information about a user. |
| UserFault (section 2.2.4.5.3) | Failure information for custom exceptions generated by a user identity provider call. |
| UserExceptionFault (section 2.2.4.5.4) | Failure information for errors generated by a user identity provider call. |
| ReturnClass (section 2.2.4.5.5) | Represents return codes for a user identity provider call |
| ProviderEndpointBehaviorAttribute (section 2.2.4.6.1) | An attribute used to decorate Provider contracts to specify their behavior. |
| ProviderInfo (section 2.2.4.6.2) | The information necessary to connect to a provider. |
| RequiredImpersonationLevelAttribute (section 2.2.4.6.3) | An attribute used to decorate Provider contracts to specify Impersonation Level. |
| PackageInfo (section 2.2.4.7.1) | The information about an add-in that is not contained in the normal metadata file. |
| ClientInstallFilter (section 2.2.4.7.2) | ~~Which~~In which clients (if any) ~~should have~~ an add-in is installed. |
| Package (section 2.2.4.7.3) | Information used to describe and install an add-in package. |
| BinaryReference (section 2.2.4.7.4) | The information needed to install an MSI on a machine. |
| Filter (section 2.2.4.7.5) | Limits which computers a client add-in MSI will be installed to. |
| OtherBinariesMap (section 2.2.4.7.6) | The list of binaries for other client platforms.<1> |
| Preinstall (section 2.2.4.7.7) | The details of preinstall. |
| UninstallConfirm (section 2.2.4.7.8) | The details of UninstallCommand. |
| VPNSetting (section 2.2.4.8.1) | The information about an add-in that is not contained in the |

| Complex Type | Description |
|---|---|
| | normal metadata file. |
| VirtualNetworkSettings (section 2.2.4.8.2) | The information about Azure Virtual Network settings |
| AddressPrefix (section 2.2.4.8.3) | An IP address range in Classless Inter-Domain Routing (CIDR) notion |
| DomainProviderCredentials (section 2.2.4.9.1) | The credentials for a domain provider. |
| DomainManagerFault (section 2.2.4.9.2) | The fault generated by a domain manager. |
| DnsRecord (section 2.2.4.9.3) | A DNS record. |
| SrvRecord (section 2.2.4.9.4) | An SRV record in DNS. |
| ARecord (section 2.2.4.9.5) | An A record in DNS. |
| MXRecord (section 2.2.4.9.6) | A mail service (MX) record in DNS. |
| TxtRecord (section 2.2.4.9.7) | A text record in DNS. |
| CNameRecord (section 2.2.4.9.8) | A canonical name record in DNS. |
| DomainState (section 2.2.4.9.9) | The status of the domain. |
| CertificateState (section 2.2.4.9.10) | The status of the certificate. |
| DomainNameConfiguration (section 2.2.4.9.11) | The currently active domain name configuration. |
| Offering (section 2.2.4.9.12) | The domain name offering. |
| DynamicDNSInfo (section 2.2.4.9.13) | The information of dynamic DNS. |
| DomainProviderManagerSettings (section 2.2.4.9.14) | The settings for Domain Manager operation. |
| O365ConfigurationException (section 2.2.4.10.1) | An error in Microsoft Office 365 integration configurations. |
| NotificationResult (section 2.2.4.11.1) | The client notification result. |
| NotificationResults (section 2.2.4.11.2) | The client notification results. |

## 2.2.4.1 Client File Backup Provider

### 2.2.4.1.1 ClientFileBackupException

The **ClientFileBackupException** type describes errors in client file backup provider.

```
        <xs:complexType name="ClientFileBackupException">
          <xs:complexContent mixed="false">
            <xs:extension xmlns:q1="http://schemas.datacontract.org/2004/07/System"
  base="q1:Exception" />
          </xs:complexContent>
        </xs:complexType>
```

## 2.2.4.1.2 ConfigurationSet

The **ConfigurationSet** type describes the global file backup configuration settings.

```xml
<xs:complexType name="ConfigurationSet">
  <xs:sequence>
    <xs:element minOccurs="0" name="FrequencyInMinutes" type="xs:unsignedInt" />
    <xs:element minOccurs="0" name="IsEnabled" type="xs:boolean" />
    <xs:element minOccurs="0" name="RetentionInMonths" type="xs:unsignedInt" />
    <xs:element minOccurs="0" name="SourceLibraries" type="tns:SourceLibrary" />
    <xs:element minOccurs="0" name="TargetPath" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="UnmanagedClients" nillable="true" type="xs:string"
/>
  </xs:sequence>
</xs:complexType>
```

**FrequencyInMinutes:** The backup frequency in minutes.

**IsEnabled:** TRUE if the client file backup is enabled; otherwise, FALSE.

**RetentionInMonths:** The backup retention age in months.

**SourceLibraries:** The source libraries to back up.

**TargetPath:** The target folder's UNC path for the caller.

**UnmanagedClients:** The list of machine names of unmanaged clients, separated by the pipe (|) character.

## 2.2.4.2  Client Backup Provider

## 2.2.4.2.1 AutoExclusionSizesInfo

The **AutoExclusionSizesInfo** type describes information about sizes of data that are automatically excluded from a backup.

```xml
<xs:complexType name="AutoExclusionSizesInfo">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="ClientSideCache" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="TemporaryFiles" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="DigitalVideoRecorder" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="PageFile" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="RecycleBin" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="HibernationFile" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="MemoryDump" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="ShadowVolumes" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="Other" type="xs:long" />
  </xs:sequence>  </xs:complexType>
```

**ClientSideCache:** The size allocated for client side cache folders.

**TemporaryFiles:** The size allocated for user temporary files.

**DigitalVideoRecorder:** The size allocated for media center temporary files.

**PageFile:** The size allocated for the system page file.

**RecycleBin:** The size allocated for the recycle bin folder.

**HibernationFile:** The size allocated for the hibernation file.

**MemoryDump:** The size allocated for the memory dump file.

**ShadowVolumes:** The size allocated for shadow volume implementation folders.

**Other:** The size allocated for other automatic exclusions.

### 2.2.4.2.2 PCBackupClientFault

The **PCBackupClientFault** type contains information about an error that occurred in the computer backup client provider service.

```
<xs:complexType name="PCBackupClientFault">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Reason" type="PCBackupClientFaultReason"
/>
    <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Other" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Reason:** The fault reason.

**Description:** The fault description.

**Other:** Any other information.

### 2.2.4.2.3 PCBackupConfiguration

The **PCBackupConfiguration** type contains all the computer backup configurations as seen from management UI.

```
<xs:complexType name="PCBackupConfiguration">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="windowStartTime" type="TimeSpan" />
    <xs:element minOccurs="1" maxOccurs="1" name="windowEndTime" type="TimeSpan" />
    <xs:element minOccurs="1" maxOccurs="1" name="dailyRetainCount" type="xs:unsignedInt"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="weeklyRetainCount" type="xs:unsignedInt"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="monthlyRetainCount" type="xs:unsignedInt"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="yearlyRetainCount" type="xs:unsignedInt"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="alertDiskBytes" type="xs:long" />
  </xs:sequence>
</xs:complexType>
```

**windowStartTime:** The backup window start time.

**windowEndTime:** The backup window end time.

**dailyRetainCount:** The daily backup retain count.

**weeklyRetainCount:** The weekly backup retain count.

**monthlyRetainCount:** The monthly backup retain count.

**yearlyRetainCount:** The yearly backup retain count.

**alertDiskBytes:** The alert disk bytes.

### 2.2.4.2.4 PCBackupServerFault

The **PCBackupServerFault** type contains information about an error that occurred in the computer backup server provider service.

```xml
<xs:complexType name="PCBackupServerFault">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Reason" type="PCBackupServerFaultReason"
/>
    <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Other" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Reason:** The fault reason.

**Description:** The fault description.

**Other:** Any other information.

### 2.2.4.2.5 PCBackupStatus

The **PCBackupStatus** type describes the state of the current backup set operation.

```xml
<xs:complexType name="PCBackupStatus">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="clientJobStatus"
      type="ClientJobStatus" />
    <xs:element minOccurs="1" maxOccurs="1" name="progress" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="failureReason"
      type="AbortReason" />
    <xs:element minOccurs="1" maxOccurs="1" name="lastBackupTime"
      type="xs:dateTime" />
    <xs:element minOccurs="0" maxOccurs="1" name="IncompleteVolumeMountPoints"
      type="ArrayOfString" />
    <xs:element minOccurs="0" maxOccurs="1" name="IncompleteVolumeLabels"
      type="ArrayOfString" />
  </xs:sequence>
</xs:complexType>
```

**clientJobStatus:** The current job status.

**progress:** How far the backup has progressed in the current phase, 0 <= progress <= 100.

**failureReason:** The reason for the most recent failure. Set only if backup status is Failed.

**lastBackupTime:** The last backup time.

**IncompleteVolumeMountPoints:** The mount points of volumes that are yet to be backed up in current backup, as well as volumes that were not backed up if the last backup failed or was incomplete.

**IncompleteVolumeLabels:** The labels of volumes that are yet to be backed up in the current backup, as well as volumes that were not backed up if the last backup failed or was incomplete.

### 2.2.4.2.6 BadFileInfo

The **BadFileInfo** type describes one file in the backup that contains a bad block.

```xml
<xs:complexType name="BadFileInfo">
```

```
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
      <xs:element minOccurs="1" maxOccurs="1" name="Size" type="xs:long" />
    </xs:sequence>
  </xs:complexType>
```

**Name:** The name of the file.

**Size:** The size of the file.

## 2.2.4.2.7 ClientConfig

The **ClientConfig** type contains all the client configuration of a backup.

```
<xs:complexType name="ClientConfig">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="IsConfigured" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="AnyVolumesIncluded" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="WindowStart" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="WindowEnd" type="xs:int" />
  </xs:sequence>
</xs:complexType>
```

**IsConfigured:** TRUE if the backup is enabled; otherwise, FALSE.

**AnyVolumesIncluded:** TRUE if any volumes eligible for backup are included; otherwise FALSE. If backup is not enabled or not configured, then the value of this variable is irrelevant.

**WindowStart:** The backup window start time, in number of minutes from midnight.

**WindowEnd:** The backup window end time, in number of minutes from midnight.

## 2.2.4.2.8 ClientConfigurationInfo

The **ClientConfigurationInfo** type describes the type of client configuration of a backup.

```
<xs:complexType name="ClientConfigurationInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Sid" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsOnline" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsConfigured" type="xs:boolean"
      />
    <xs:element minOccurs="1" maxOccurs="1" name="AllVolumesConfigured"
      type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="AnyVolumesIncluded"
      type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="JobsExist" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="LastGoodBackupTime"
      type="xs:dateTime" />
    <xs:element minOccurs="1" maxOccurs="1" name="LastBackupEnabledTime"
      type="xs:dateTime" />
    <xs:element minOccurs="1" maxOccurs="1" name="LastStatus"
      type="ClientJobStatus" />
  </xs:sequence>
</xs:complexType>
```

The following members SHOULD be used only for reading the property. Each is marked as writable only so that it can support the infrastructure.

**Sid:** The client machine SID, used to uniquely identify the client.

**Name:** The client machine name.

**IsOnline:** TRUE if the machine is online; otherwise FALSE.

**IsConfigured:** TRUE if client is configured for backup; otherwise FALSE.

**AllVolumesConfigured:** TRUE if all volumes are configured for backup; otherwise FALSE.

**AnyVolumesIncluded:** TRUE if at least one volume is included for backup; otherwise FALSE.

**JobExist:** TRUE if a backup exists for the client; otherwise false.

**LastGoodBackupTime:** The time of the last successfully completed backup.

**LastBackupEnabledTime:** When the client was last enabled for backups.

**LastStatus:** The last backup status.

### 2.2.4.2.9 ClientJobInfo

The **ClientJobInfo** type describes one machine-wide backup job.

```
<xs:complexType name="ClientJobInfo">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Index" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="StartTime" type="xs:dateTime" />
    <xs:element minOccurs="1" maxOccurs="1" name="Duration" type="TimeSpan" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesUsed" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsAutomatic" type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="Status" type="ClientJobStatus" />
    <xs:element minOccurs="1" maxOccurs="1" name="retentionState" type="RetentionState" />
    <xs:element minOccurs="1" maxOccurs="1" name="failureReason" type="AbortReason" />
    <xs:element minOccurs="0" maxOccurs="1" name="FailureDetails" type="ArrayOfString" />
  </xs:sequence>
</xs:complexType>
```

The following members SHOULD be used only for reading the property. Each is marked as writable only so that it can support the infrastructure.

**Index:** The backup job index.

**StartTime:** The backup start time.

**Duration:** The duration for which the backup ran.

**BytesUsed:** The bytes used on disk on the server for this backup.

**IsAutomatic:** TRUE if a scheduled backup; FALSE if manual.

**Description:** The backup job description.

**Status:** The backup job status.

**retentionState:** The retention state for the job, such as "marked for deletion", following the normal retention policy, or locked status.

**failureReason:** The failure reason for the job, or "none" if the job succeeded.

**FailureDetails:** The failure details for the job. Empty array or null if successful.

### 2.2.4.2.10    ClientRepairStatus

The **ClientRepairStatus** type describes the status of backups for each client after repair.

```xml
<xs:complexType name="ClientRepairStatus">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Sid" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="DataStatus" type="PCClientDataStatus" />
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

The following members SHOULD be used only for reading the property. Each is marked as writable only so that it can support the infrastructure.

**Sid:** The client SID, used to uniquely identify the client.

**DataStatus:** The status of backup for this client.

**Name:** The client name, used to identify a client to the user.

### 2.2.4.2.11    ClientVolumeInfo

The **ClientVolumeInfo** type describes a one-client volume.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:complexType name="ClientVolumeInfo">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="guid"
      xmlns:q1="http://microsoft.com/wsdl/types/" type="q1:guid" />
    <xs:element minOccurs="0" maxOccurs="1" name="uniqueIdBytes"
      type="ArrayOfUnsignedByte" />
    <xs:element minOccurs="0" maxOccurs="1" name="MountPoint" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="VolumeName" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="Capacity" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesUsed" type="xs:long" />
    <xs:element minOccurs="0" maxOccurs="1" name="FileSystem" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesPerSector" type="xs:int"
      />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesPerCluster" type="xs:int"
      />
    <xs:element minOccurs="1" maxOccurs="1" name="Location" type="VolumeLocation"
      />
    <xs:element minOccurs="1" maxOccurs="1" name="BackupBytes" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="Status" type="VolumeStatus" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsEligibleForBackup"
      type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="JobsExist" type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="ExcludedFolders"
      type="ArrayOfString" />
  </xs:sequence>
</xs:complexType>
```

The following members SHOULD be used only for reading the property. Each is marked as writable only so that it can support the infrastructure.

**guid:** The volume **GUID**.

**uniqueIdBytes:** The unique ID of the volume.

**MountPoint:** The volume mount point.

**VolumeName:** The name of the volume.

**Capacity:** The total capacity of the volume in bytes.

**BytesUsed:** The total number of bytes used on the volume.

**FileSystem:** The file system type of the volume.

**BytesPerSector:** The bytes per sector for the volume.

**BytesPerCluster:** The bytes per cluster for the volume.

**Location:** The volume location details, such as internal, external, and so on.

**BackupBytes:** The total number of bytes for backup on the volume.

**Status:** The volume status.

**IsEligibleForBackup:** TRUE if volume is eligible for backup; otherwise FALSE.

**JobsExist:** TRUE if any volume jobs exist for the volume; otherwise FALSE.

**ExcludedFolders:** A list of excluded folders.

### 2.2.4.2.12    ConfigureBackupFault

The **ConfigureBackupFault** type contains information about an error that occurred when configure backup for a client failed due to another configure backup being in progress.

```
<xs:complexType name="ConfigureBackupFault">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Reason"
      type="ConfigureBackupFaultReason" />
    <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string"
      />
    <xs:element minOccurs="0" maxOccurs="1" name="Other" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Reason:** The fault reason.

**Description:** The fault description.

**Other:** Any other information, such as Exception Stack trace if needed for logging.

### 2.2.4.2.13    ExcludedFolderInfo

The **ExcludedFolderInfo** type describes one excluded folder.

```
<xs:complexType name="ExcludedFolderInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="Size" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsAutomatic" type="xs:boolean"
      />
  </xs:sequence>
</xs:complexType>
```

**Name:** The folder name.

**Size:** The folder size, excluding automatically excluded content.

**IsAutomatic:** TRUE, if and only if the folder is automatically excluded; otherwise FALSE.

### 2.2.4.2.14 FolderInfo

The **FolderInfo** type describes one folder.

```
<xs:complexType name="FolderInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="FileRecordIndex" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="UsedBytes" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="ExclusionReason" type="ExclusionReason"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="IsFolder" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="HasChildren" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
```

**Name:** The folder name.

**FileRecordIndex:** The file record index.

**UsedBytes:** The bytes used by this folder, including subfolders.

**ExclusionReason:** The exclusion reason.

**IsFolder:** TRUE only if the entry is a folder; otherwise FALSE.

**HasChildren:** TRUE, if this folder has children; otherwise, FALSE.

### 2.2.4.2.15 FolderInfoGroup

The **FolderInfoGroup** type describes a group of folders.

```
<xs:complexType name="FolderInfoGroup">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="FolderInfoList" type="ArrayOfFolderInfo"
/>
    <xs:element minOccurs="1" maxOccurs="1" name="TotalNumChildren" type="xs:int" />
  </xs:sequence>
</xs:complexType>
```

**FolderInfoList:** The list of folder information objects in the group.

**TotalNumChildren:** The total number of children in the group.

### 2.2.4.2.16 FolderRequest

The **FolderRequest** type describes the data needed for Get Folder Children.

```
<xs:complexType name="FolderRequest">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="volumeGuid"
xmlns:q1="http://microsoft.com/wsdl/types/" type="q1:guid" />
    <xs:element minOccurs="1" maxOccurs="1" name="fileRecordIndex" type="xs:long" />
  </xs:sequence>
</xs:complexType>
```

**volumeGuid:** The GUID of the volume.

**fileRecordIndex:** The index of the file record.

### 2.2.4.2.17 OperationStatusResponse

The **OperationStatusResponse** type contains the response of a Get Operation Status request.

```xml
<xs:complexType name="OperationStatusResponse">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Operation" type="PCBackupStatusOperation"
/>
    <xs:element minOccurs="0" maxOccurs="1" name="ClientSid" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Machine" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="PercentDone" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="Phase" type="PCBackupVolumePhase" />
    <xs:element minOccurs="1" maxOccurs="1" name="RepairState" type="PCBackupRepairState"
/>
  </xs:sequence>
</xs:complexType>
```

**Operation:** The type of operation in progress.

**ClientSid:** The GUID of the client backing up or restoring, or Guid.Empty if none.

**Machine:** The name of the client backing up or restoring, or null if none.

**PercentDone:** The progress indicator of the current operation, from 0 to 100 percent.

**Phase:** The phase of the operation.

**RepairState:** The repair state of the operation.

### 2.2.4.2.18 ServiceStatusResponse

The **ServiceStatusResponse** type contains the response of a service status request.

```xml
<xs:complexType name="ServiceStatusResponse">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Ready" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
```

**Ready:** TRUE only if the service is ready for requests; otherwise FALSE.

### 2.2.4.2.19 VolumeJobInfo

The **VolumeJobInfo** type contains information about a volume job.

```xml
<xs:complexType name="VolumeJobInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="volumeInfo" type="ClientVolumeInfo" />
    <xs:element minOccurs="0" maxOccurs="1" name="uniqueIdBytes" type="xs:base64Binary" />
    <xs:element minOccurs="0" maxOccurs="1" name="MountPoint" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="VolumeName" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="FileSystem" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesPerSector" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesPerCluster" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="StartTimeUtc" type="xs:dateTime" />
    <xs:element minOccurs="1" maxOccurs="1" name="Duration" type="TimeSpan" />
    <xs:element minOccurs="1" maxOccurs="1" name="InitialCapacity" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="MinimumCapacity" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="BytesUsed" type="xs:long" />
    <xs:element minOccurs="1" maxOccurs="1" name="Status" type="VolumeJobStatus" />
```

```
              <xs:element minOccurs="0" maxOccurs="1" name="autoExclusionSizes"
    type="AutoExclusionSizesInfo" />
              <xs:element minOccurs="0" maxOccurs="1" name="ExcludedFolders"
    type="ArrayOfExcludedFolderInfo" />
              <xs:element minOccurs="0" maxOccurs="1" name="BadFiles" type="ArrayOfBadFileInfo" />
              <xs:element minOccurs="1" maxOccurs="1" name="failureReason" type="AbortReason" />
              <xs:element minOccurs="0" maxOccurs="1" name="FailureDetails" type="ArrayOfString" />
          </xs:sequence>
        </xs:complexType>
```

The following members SHOULD be used only for reading the property. Each is marked as writable so as to support the infrastructure.

**volumeInfo:** The volume details.

**uniqueIdBytes:** The unique ID of the volume at the time of backup.

**MountPoint:** The volume mount point at time of backup.

**VolumeName:** The volume name at time of backup.

**FileSystem:** The file system time at the time of backup.

**BytesPerSector:** The bytes per sector for the volume at the time of backup.

**BytesPerCluster:** The bytes per cluster for the volume at the time of backup.

**StartTimeUtc:** The **Coordinated Universal Time (UTC)** start time for the volume job.

**Duration:** The volume job duration.

**InitialCapacity:** The initial capacity of the volume during backup, in bytes.

**MinimumCapacity:** The minimum capacity of the volume required for restoring this volume backup.

**BytesUsed:** The bytes used for this volume job.

**Status:** The volume job status for the volume job.

**autoExclusionSizes:** The size of all auto-excluded folders.

**ExcludedFolders:** List of excluded folders.

**BadFiles:** Information about the file that contains a list of bad blocks on the volume.

**failureReason:** The failure reason for the volume job. If the job succeeded, this element SHOULD be set to "None".

**FailureDetails:** The failure details for the volume job. If the job succeeded, this element SHOULD be set to Null or an empty array.

### 2.2.4.2.20    WaitingOperation

The **WaitingOperation** type contains information about a waiting operation.

```
    <xs:complexType name="WaitingOperation">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="Operation" type="PCBackupStatusOperation"
    />
        <xs:element minOccurs="0" maxOccurs="1" name="MachineSID" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="MachineName" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="PercentComplete" type="xs:int" />
      </xs:sequence>
```

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

**Operation:** An indicator that the backup operation is queued.

**MachineSID:** The SID of the machine for which backup/restore is queued.

**MachineName:** The name of the machine for which backup/restore is queued.

**PercentComplete:** The progress indicator of the current operation, from 0 to 100 percent

### 2.2.4.2.21      RepairResult

The **RepairResult** type contains information about the result of a repair job.

```
    <xs:complexType name="RepairResult">
      <xs:sequence>
        <xs:element minOccurs="0" name="ClientStatus" nillable="true"
  type="tns:ArrayOfClientRepairStatus" />
        <xs:element minOccurs="0" name="RepairStartTime" type="xs:dateTime" />
        <xs:element minOccurs="0" name="RepairState" type="tns:PCBackupRepairState" />
      </xs:sequence>
    </xs:complexType>
```

**ClientStatus:** A list of client repair status values.

**RepairStartTime:** The start time of the repair job.

**RepairState:** The state of the repair job.

### 2.2.4.3  Health Provider

### 2.2.4.3.1 AlertListChunk

The **AlertListChunk** type describes a portion of a list of alerts.

```
    <xs:complexType name="AlertListChunk">
      <xs:sequence>
        <xs:element minOccurs="0" name="Alerts" nillable="true"
  xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
  lth.AlertFramework" type="q1:ArrayOfAlertInfo" />
        <xs:element minOccurs="0" name="HasMore" type="xs:boolean" />
      </xs:sequence>
    </xs:complexType>
```

**Alerts:** The alerts in this chunk.

**HasMore:** TRUE, if there are more alerts to be sent in another chunk; otherwise FALSE.

### 2.2.4.3.2 AlertInfo

The **AlertInfo** type contains informational data about a Network Health Alert.

```
    <xs:complexType name="AlertInfo">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="InstanceId" type="Guid" />
        <xs:element minOccurs="1" maxOccurs="1" name="MachineName" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="MachineId" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="FeatureName" type="xs:string" />
```

```
        <xs:element minOccurs="1" maxOccurs="1" name="HealthDefinitionName" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="FeatureTitle" type=" xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="HealthDefinitionTitle" type="xs:string"
/>
        <xs:element minOccurs="1" maxOccurs="1" name="Description" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Severity" type="HealthStatus" />
        <xs:element minOccurs="1" maxOccurs="1" name="ActionPerformed" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="IsRepairDefined" type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="RepairActionPerformed" type="xs:string"
/>
        <xs:element minOccurs="1" maxOccurs="1" name="DateAndTime" type="xs:dateTime" />
        <xs:element minOccurs="1" maxOccurs="1" name="IsSystem" type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="TroubleshootSteps" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="SupportLink" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Troubleshooter" type="UserContextTask" />
        <xs:element minOccurs="1" maxOccurs="1" name="IsNew" type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="IsSuppressed" type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="LastSuppressedDateTime"
type="xs:dateTime" />
        <xs:element minOccurs="1" maxOccurs="1" name="AdditionalInfo"
type="IEnumerable<string>" />
        <xs:element minOccurs="1" maxOccurs="1" name="Escalate" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
```

**InstanceId**: The instance ID of the alert.

**MachineName**: The name of the machine on which the alert is raised.

**MachineId**: Typically the SID of the client computer.

**FeatureName**: The name of the feature definition to which the health definition belongs.

**HealthDefinitionName**: The name of the feature definition for which the alert is raised.

**FeatureTile**: The title of the health definition for which the alert is raised.

**HealthDefinitionTitle**: The title of the feature definition to which the health definition belongs.

**Description**: A description of the alert.

**Severity**: The health status.

**ActionPerformed**: The health definition alert action performed when the alert is detected.

**IsRepairDefined**: TRUE, if the alert is repairable; otherwise, FALSE.

**RepairActionPerformed**: The information about the health definition repair action performed when the alert is repaired.

**DateAndTime**: The date and time when the alert was raised.

**IsSystem**: TRUE, if the alert is a system alert; otherwise, FALSE.

**TroubleshootSteps**: The troubleshooting steps of the alert.

**SupportLink**: The support link of the alert.

**Troubleshooter**: Allows the user to launch the task from the Alert Viewer.

**IsNew**: TRUE, if the alert being raised is not already present; otherwise, FALSE.

**IsSuppressed**: TRUE, if the alert is suppressed; otherwise, FALSE.

**LastSuppressedDateTime**: The last time the alert was suppressed.

**AdditionalInfo**: A store of additional information bound to the alert.

**Escalate**: TRUE, if the alert will be escalated to the administrator through email; otherwise, FALSE.

### 2.2.4.3.3 UserContextTask

The **UserContextTask** type defines a task that is expected to be launched from a user context.

```
<xs:complexType name="UserContextTask">
  <xs:sequence>
    <xs:element minOccurs="0" name="Arguments" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Program" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Title" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Arguments:** The command line arguments to the program. These arguments can contain references to environment variables.

**Program:** The program path of the task. This path can contain references to environment variables.

**Title:** The title of the task.

### 2.2.4.3.4 ClearCommand

The **ClearCommand** type describes the command to clear an alert.

```
<xs:complexType name="ClearCommand">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="AlertInstanceId" type="Guid" />
    <xs:element minOccurs="1" maxOccurs="1" name="MachineId" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="AlertDateAndTime" type="xs:dateTime" />
  </xs:sequence>
</xs:complexType>
```

**AlertInstanceId**: The instance SID of the alert.

**MachineId**: The SID of the target machine.

**AlertDateAndTime**: The alert date and time.

### 2.2.4.3.5 RaiseCommand

The **RaiseCommand** type describes the command to raise an alert.

```
<xs:complexType name="RaiseCommand">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="RaisedAlert" type="AlertInfo" />
  </xs:sequence>
</xs:complexType>
```

**RaisedAlert**: The alert that needs to be raised.

### 2.2.4.3.6 SuppressCommand

The **SuppressCommand** type describes the command to suppress an alert.

```xml
<xs:complexType name=" SecurityUpdateDetailProperty">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="AlertInstanceId" type="Guid" />
    <xs:element minOccurs="1" maxOccurs="1" name="MachineId" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="AlertDateAndTime" type="xs:dateTime" />
    <xs:element minOccurs="1" maxOccurs="1" name="Suppress" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="SuppressTime" type="xs:dateTime" />
  </xs:sequence>
</xs:complexType>
```

**AlertInstanceId**: The instance SID of the alert.

**MachineId**: The SID of the target machine.

**AlertDateAndTime**: The alert date and time.

**Suppress**: TRUE, if the user wants to suppress an alert; otherwise, FALSE.

**SuppressTime**: The date and time when the alert was suppressed.

## 2.2.4.4  Machine Identity Provider

### 2.2.4.4.1 ReturnClass Type

The **ReturnClass** type describes the return class for the Machine Identity Provider.

```xml
<xs:complexType name="ReturnClass">
  <xs:sequence>
    <xs:element minOccurs="0" name="ErrorCatalog" type="tns:ErrorCatalogType" />
    <xs:element minOccurs="0" name="NativeError" type="xs:int" />
  </xs:sequence>
</xs:complexType>
```

**ErrorCatalog:** The ErrorCatalogType value of an error, if one occurred.

**NativeError:** The native error code if an error happened.

### 2.2.4.4.2 GetMachineStatusType

The **GetMachineStatusType** type contains the result of GetMachineStatus request.

```xml
<xs:complexType name="GetMachineStatusType">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="MaxClientComputer" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="CurrentClientComputer" type="xs:int" />
    <xs:element minOccurs="1" maxOccurs="1" name="MachineStatus" type="MachineStatus" />
  </xs:sequence>
</xs:complexType>
```

**MaxClientComputer**: The maximum number of clients allowed.

**CurrentClientComputer**: The current number of clients.

**MachineStatus**: The return status for the specified client machine.

## 2.2.4.5  User Identity Provider

## 2.2.4.5.1 PersonalInfo

The **PersonalInfo** type describes user personal information.

```
<xs:complexType name="PersonalInfo">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="IsLocalUser" type="xs:boolean" />
    <xs:element minOccurs="0" maxOccurs="1" name="UserFolder" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="FirstName" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="LastName" type="xs:string" />
    <xs:element minOccurs="1" maxOccurs="1" name="RemoteAccess" type="RemoteAccessType" />
    <xs:element minOccurs="1" maxOccurs="1" name="VpnAccess" type="RemoteAccessType" />
    <xs:element minOccurs="1" maxOccurs="1" name="AccessLevel" type="AccessLevelType" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsBuiltIn" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="IsShowInUI" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
```

**IsLocalUser**: TRUE, if this is the local user; otherwise, FALSE.

**UserFolder**: The path of the shared folder belonging to this user.

**FirstName**: The first name of the user.

**LastName**: The last name of the user.

**RemoteAccess**: The remote access status of the user.

**VpnAccess**: The VPN access status of the user.

**AccessLevel**: The access level of the user.

**IsBuiltIn**: TRUE, if the user is built-in; otherwise, FALSE.

**IsShowInUI**: TRUE, if user will be shown in the UI; otherwise, FALSE.

## 2.2.4.5.2 UserInfo

The **UserInfo** type presents the information of a user.

```
<xs:complexType name="UserInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="UserName" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="PersonalInfo" type="PersonalInfo" />
    <xs:element minOccurs="1" maxOccurs="1" name="UserStatus" type="UserStatus" />
    <xs:element minOccurs="1" maxOccurs="1" name="Warning" type="UserCustomExceptionCode"
/>
    <xs:element minOccurs="0" maxOccurs="1" name="WindowsSID" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="WindowsAccount" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="FullUserName" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**UserName**: The name of the user.

**PersonalInfo**: The detailed personal information of the given user.

**UserStatus**: The status of the user.

**Warning**: The last error of import/update/add user.

**WindowsSID**: The SID of this user.<2>

**WindowsAccount**: The account of this user.<3>

**FullUserName**: The domain/server name of the user.

### 2.2.4.5.3 UserFault

The **UserFault** type contains failure information from a custom user identity provider call.

```
<xs:complexType name="UserFault">
  <xs:sequence>
    <xs:element minOccurs="0" name="ExceptionCode" type="xs:int" />
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="UserCode" type="tns:UserCustomExceptionCode" />
  </xs:sequence>
</xs:complexType>
```

**ExceptionCode:** The custom exception code.

**Message:** The custom error message.

**UserCode:** The user custom exception code.

### 2.2.4.5.4 UserExceptionFault

The **UserExceptionFault** type contains information about exceptions raised by a custom user identity provider call.

```
<xs:complexType name="UserExceptionFault">
  <xs:sequence>
    <xs:element minOccurs="0" name="ErrorCode" type="xs:int" />
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="UserErrorCode" type="tns:UserCustomExceptionCode" />
  </xs:sequence>
</xs:complexType>
```

**ErrorCode**: The custom error code.

**Message:** The custom error message.

**UserErrorCode:** The user custom exception code.

### 2.2.4.5.5 ReturnClass

The **ReturnClass** type represents return codes for a custom user identity provider call.

```
<xs:complexType name="ReturnClass">
  <xs:sequence>
    <xs:element minOccurs="0" name="NativeError" type="xs:int" />
    <xs:element minOccurs="0" name="UserCode" type="tns:UserCustomExceptionCode" />
  </xs:sequence>
</xs:complexType>
```

**NativeError:** The native error code.

**UserCode:** The user custom exception code.

### 2.2.4.6  Provider Registry Service

### 2.2.4.6.1 ProviderEndpointBehaviorAttribute

The **ProviderEndpointBehaviorAttribute** type describes the attribute to use to decorate Provider contracts to specify their behavior.

```
   <xs:complexType name="ProviderEndpointBehaviorAttribute">
     <xs:sequence>
       <xs:element minOccurs="1" maxOccurs="1" name="EndpointCredentialType"
type="CredentialType" />
       <xs:element minOccurs="0" maxOccurs="1" name="AllowedConnectionType"
type="ConnectionSetting" />
     </xs:sequence>
   </xs:complexType>
```

**EndpointCredentialType:** The credential type expected by a provider.

**AllowedConnectionType:** The connection setting used or expected by a provider.

### 2.2.4.6.2 ProviderInfo

The **ProviderInfo** type represents the information necessary to connect to a provider.

```
   <xs:complexType name="ProviderInfo">
     <xs:sequence>
       <xs:element minOccurs="0" name="Address" nillable="true" type="xs:anyURI" />
       <xs:element minOccurs="0" name="Binding" nillable="true" type="xs:string" />
       <xs:element minOccurs="0" name="Connected" type="xs:boolean" />
       <xs:element minOccurs="0" name="Contract" nillable="true" type="xs:string" />
       <xs:element minOccurs="0" name="Host" nillable="true" type="xs:string" />
       <xs:element minOccurs="0" name="Identifier" nillable="true" type="xs:string" />
       <xs:element minOccurs="0" name="IsInProc" type="xs:boolean" />
       <xs:element minOccurs="0" name="ProviderEndpointBehavior" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Common.Pro
viderFramework" type="q1:ProviderEndpointBehaviorAttribute" />
       <xs:element minOccurs="0" name="RequiredImpersonationLevel" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Common.Pro
viderFramework" type="q2:RequiredImpersonationLevelAttribute" />
     </xs:sequence>
   </xs:complexType>
```

**Address:** The **URI** of the provider.

**Binding:** The binding type to use for the provider.

**Connected:** TRUE, if the provider is currently connected; otherwise, FALSE.

**Contract:** The name of the contract of the provider, as a string.

**AddressHost:** The **URI**host of ~~the~~this provider.

**Binding:** ~~The binding type to use for the provider.~~

**ProviderEndpointBehavior:** ~~The endpoint behavior of the provider.~~

**RequiredImpersonationLevel:** ~~The required client's impersonation level needed by the provider.~~

**Identifier:** The identifier of the provider. The identifier is used to differentiate between multiple providers that expose the same contract.

**IsInProc:** True if the provider is in-process loadable; otherwise, false.

~~**Host**~~**ProviderEndpointBehavior:** The ~~host~~endpoint behavior of ~~this~~the provider.

**RequiredImpersonationLevel:** The required client's impersonation level needed by the provider.

### 2.2.4.6.3 RequiredImpersonationLevelAttribute

The **RequiredImpersonationLevelAttribute** type describes the attribute to use to decorate provider contracts to specify the Impersonation Level.

```
<xs:complexType name="RequiredImpersonationLevelAttribute">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Level" type="ImpersonationLevel" />
  </xs:sequence>
</xs:complexType>
```

**Level:** The impersonation level expected by the provider.

### 2.2.4.7 Addin Management

### 2.2.4.7.1 PackageInfo

The **PackageInfo** type describes the information about an add-in that is not contained in the typical metadata file.

```
<xs:complexType name="PackageInfo">
    <xs:sequence>
        <xs:element minOccurs="0" name="Company" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="ExpirationDateUtc" type="xs:dateTime" />
        <xs:element minOccurs="0" name="Filter" nillable="true"
type="tns:ClientInstallFilter" />
        <xs:element minOccurs="0" name="Install" type="xs:boolean" />
        <xs:element minOccurs="0" name="InstallDateUtc" type="xs:dateTime" />
        <xs:element minOccurs="0" name="LastRedeployDate" type="xs:dateTime" />
        <xs:element minOccurs="0" name="Package" nillable="true" type="tns:Package" />
        <xs:element minOccurs="0" name="RenewalUri" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="UpdateType" type="tns:UpdateClassification" />
        <xs:element minOccurs="0" name="UpdateUri" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="UpdateVersion" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/System" type="q1:Version" />
        <xs:element minOccurs="0" name="VersionOnServer" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/System" type="q2:Version" />
    </xs:sequence>
</xs:complexType>
```

**Company:** The company that the add-in package is for. This is set only for add-ins that have a verified digital signature.

**ExpirationDateUtc:** The date on which the add-in package subscription will expire. DateTime.MinValue means that there is no expiration date.

**Filter:** Determines which clients (if any) to install the add-in to.

**Install:** TRUE, if the add-in is meant to be installed; otherwise, FALSE.

**InstallDateUtc:** The time the add-in was first installed.

**LastRedeployDate:** The date of the last redeployment or of the initial install if there was not a redeploy.

**Package:** The actual add-in metadata.

**RenewalUri:** The address to show for renewing the add-in.

**UpdateType:** The type of update.

**UpdateUri:** The Uri of the latest update.

**UpdateVersion:** The version of the latest update.

**VersionOnServer:** The version of the package that is currently installed on the server. Used to prevent clients from installing/upgrading before the server has installed the package.

### 2.2.4.7.2 ClientInstallFilter

The **ClientInstallFilter** type indicates in which clients (if any) ~~should have~~ an add-in is installed.

```
<xs:complexType name="ClientInstallFilter">
    <xs:sequence>
        <xs:element minOccurs="0" name="clientSidList" nillable="true"
xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q3:ArrayOfstring"
/>
        <xs:element minOccurs="0" name="listIsMachinesToInclude" type="xs:boolean" />
    </xs:sequence>
</xs:complexType>
```

**clientSIDList:** The list of client machine SIDs, will always be sorted.

**listIsMachinesToInclude:** TRUE, if the list is of machines to be included; otherwise, FALSE.

### 2.2.4.7.3 Package

The **Package** type contains information used to describe and install an add-in package.

```
<xs:complexType name="Package">
    <xs:sequence>
        <xs:element name="Id" type="ser:guid" />
        <xs:element name="Version" nillable="true" type="xs:string" />
        <xs:element name="Name" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="Allow32BitOn64BitClients" type="xs:boolean" />
        <xs:element minOccurs="0" name="ServerBinary" nillable="true"
type="tns:BinaryCollection" />
        <xs:element minOccurs="0" name="ClientBinary32" nillable="true"
type="tns:BinaryCollection" />
        <xs:element minOccurs="0" name="ClientBinary64" nillable="true"
type="tns:BinaryCollection" />
        <xs:element minOccurs="0" name="SupportedSkus" nillable="true"
type="tns:SkuCollection" />
        <xs:element minOccurs="0" name="SupportUrl" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="Location" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="PrivacyStatement" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="OtherBinaries" nillable="true"
type="tns:OtherBinariesMap" />
        <xs:element minOccurs="0" name="Preinstall" nillable="true" type="tns:Preinstall" />
        <xs:element minOccurs="0" name="UninstallConfirm" nillable="true"
type="tns:UninstallConfirm" />
    </xs:sequence>
</xs:complexType>
```

**Id:** The ID for the package.

**Version:** The version of the add-in package.

**Name:** The name of the package.

**Allow32BitOn64BitClients:** TRUE, if 64-bit clients ~~should~~ fall back to the 32-bit binaries ~~if~~when no 64-bit binaries can be found; otherwise, FALSE.

**ServerBinary:** The binaries to install on the server.

**ClientBinary32:** The binary to install on 32-bit clients.

**ClientBinary64:** The binary to install on 64-bit clients.

**SupportedSkus:** A list of operating systems that are supported by this package.

**SupportUrl:** The support URL for add-in.

**Location:** Indicates whether the add-in integrates with an online service or mainly has functionality local to the server.

**PrivacyStatement:** A link to the privacy statement for the add-in.

**OtherBinaries:** Other client binaries.<4>

**Preinstall:** The specified preinstall command.

**UninstallConfirm:** The specified uninstall command to override the default action.

### 2.2.4.7.4 BinaryReference

The **BinaryReference** type encapsulates the information needed to install an MSI on a machine.

```
<xs:complexType name="BinaryReference">
  <xs:sequence>
    <xs:element name="ProductCode" type="ser:guid" />
    <xs:element name="Version" nillable="true" type="xs:string" />
    <xs:element name="Name" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Arguments" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Filter" nillable="true" type="tns:Filter" />
  </xs:sequence>
</xs:complexType>
```

**ProductCode:** The product code for the MSI to be installed.

**Version:** The version of the MSI to be installed.

**Name:** The name of the file to be installed.

**Arguments:** Any parameters that will be used to install the MSI.

**Filter:** The filter used to determine whether the binary applies to a given computer.

### 2.2.4.7.5 Filter

The Filter type limits which client computers the add-in MSI can be installed on.

```
<xs:complexType name="Filter">
  <xs:sequence>
    <xs:element minOccurs="0" name="MinOSVersion" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="MaxOSVersion" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="OSEdition"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.AddinInfra
structure" type="q1:OSEditions" />
    <xs:element minOccurs="0" name="Language" nillable="true" type="xs:string" />
```

```
            <xs:element minOccurs="0" name="IsLanguageFallback" type="xs:boolean" />
        </xs:sequence>
    </xs:complexType>
```

**MinOSVersion:** The minimum OS version supported (inclusive).

**MaxOSVersion:** The maximum OS version supported (inclusive).

**OSEdition:** The supported OS editions.

**Language:** The supported OS languages.

**IsLanguageFallback**: TRUE, if this binary is to be used as the language fallback when the client OS language is not used as the language fallback; otherwise, false.

### 2.2.4.7.6 OtherBinariesMap

The **OtherBinariesMap** type describes the list of binaries for other client platforms.<5>

```
    <xs:complexType name="OtherBinariesMap">
      <xs:annotation>
        <xs:appinfo>
        <IsDictionary
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">true</IsDictionary>
        </xs:appinfo>
      </xs:annotation>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="Item">
          <xs:complexType>
            <xs:sequence>
            <xs:element name="Target" nillable="true" type="xs:string" />
            <xs:element name="FileName" nillable="true" type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
```

**Target:** The key of the map.

**FileName:** The value of the map.

### 2.2.4.7.7 Preinstall

The **Preinstall** type describes the details of preinstall.

```
    <xs:complexType name="Preinstall">
      <xs:sequence>
        <xs:element name="Executable" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="NormalArgs" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="SilentArgs" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="IgnoreExitCode" type="xs:boolean" />
      </xs:sequence>
    </xs:complexType>
```

**Executable:** The file name of executable; the path is relative inside the package.

**NormalArgs:** An argument for normal option.

**SilentArgs:** An argument for silent option.

**IgnoreExitCode:** Whether to ignore the exit code.

### 2.2.4.7.8 UninstallConfirm

The **UninstallConfirm** type describes the details of UninstallCommand.

```
<xs:complexType name="UninstallConfirm">
  <xs:sequence>
    <xs:element name="Executable" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Arguments" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Executable:** The file name of the executable.

**Arguments:** The arguments of the executable.

### 2.2.4.8  Network Service

### 2.2.4.8.1 VPNSetting

The **VPNSetting** type describes the information about an add-in that is not contained in the normal metadata file.

```
<xs:complexType name="VPNSetting">
  <xs:sequence>
    <xs:element minOccurs="0" name="ConnectionName" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="InternetAddress" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="IntranetDnsSuffix" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**ConnectionName:** The name for the VPN connection (that is, both the phone-book entry name and the PPP adapter name).

**InternetAddress:** The connect-to address for the VPN connection. It ~~should~~can be either a public IP address or a **fully qualified domain name (FQDN)** that can be resolved on the Internet. VPN clients use this address to dial-up the VPN server.

**IntranetDnsSuffix:** The common suffix of the FQDNs of the computers on the private network, which will be set as the connection-specific DNS suffix of the PPP adapter on VPN clients. With these settings, nondomain-joined clients are able to resolve FQDNs of the computers on the private network.

### 2.2.4.8.2 VirtualNetworkSettings

~~Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

The **VirtualNetworkSettings** type SHOULD<6>~~describes~~ describe the Azure Virtual Network settings.

```
<xs:complexType name="VirtualNetworkSettings">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Enabled" type="xs:boolean" />
```

```
        <xs:element minOccurs="0" maxOccurs="1" name="VpnDeviceName" nillable="true"
type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="VirtualNetworkAddressPrefixes"
nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Networking
" type="q1:ArrayOfAddressPrefix " />
    </xs:sequence>
</xs:complexType>
```

**Enabled:** TRUE if the Azure Virtual Network integration is enabled; otherwise, FALSE.

**VpnDeviceName:** The name of the local VPN device.

**VirtualNetworkAddressPrefixes:** The IP address ranges assigned to the Azure Virtual Network resources in Classless Inter-Domain Routing (CIDR) notation.

### 2.2.4.8.3 AddressPrefix

~~Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

The **AddressPrefix** type SHOULD<7>~~describes~~ describe an IP address range in Classless Inter-Domain Routing (CIDR) notation. For more information about Classless Inter-Domain Routing (CIDR), see [RFC1518] and [RFC1519].

```
<xs:complexType name="AddressPrefix">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Address" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/System.Net" type="q2:IPAddress"/>
    <xs:element minOccurs="1" maxOccurs="1" name="PrefixLength" type="xs:int" />
  </xs:sequence>
</xs:complexType>
```

**Address:** The IP address of the Classless Inter-Domain Routing (CIDR) block.

**PrefixLength:** The prefix length of the Classless Inter-Domain Routing (CIDR) block.

### 2.2.4.9 Domain Infrastructure Service

### 2.2.4.9.1 DomainProviderCredentials

The **DomainProviderCredentials** type carries credentials for the domain provider.

```
<xs:complexType name="DomainProviderCredentials">
  <xs:sequence>
    <xs:element minOccurs="0" name="Password" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="UserName" nillable="true" type="xs:string" />
  </xs:sequence>
</xs:complexType>
```

**Password:** The password for the credential.

**UserName:** The user name for the credential.

### 2.2.4.9.2 DomainManagerFault

The **DomainManagerFault** type describes the fault generated by the domain manager.

```
<xs:complexType name="DomainManagerFault">
  <xs:sequence>
    <xs:element minOccurs="0" name="Detail" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Message" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="Reason" type="tns:FailureReason" />
  </xs:sequence>
</xs:complexType>
```

**Detail:** The details about this fault.

**Message:** A message about this fault.

**Reason:** The reason for this fault.

### 2.2.4.9.3 DnsRecord

The **DnsRecord** type describes a DNS record.

```
<xs:complexType name="DnsRecord">
  <xs:sequence>
    <xs:element minOccurs="0" name="Label" nillable="true" type="xs:string" />
    <xs:element minOccurs="0" name="TimeToLive" type="ser:duration" />
  </xs:sequence>
</xs:complexType>
```

**Label:** The label of the record.

**TimeToLive:** The time span within which the record is effective.

### 2.2.4.9.4 SrvRecord

The **SrvRecord** type describes a **service (SRV) resource record** in **DNS**.

```
<xs:complexType name="SrvRecord">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DnsRecord">
      <xs:sequence>
        <xs:element minOccurs="0" name="Port" type="xs:int" />
        <xs:element minOccurs="0" name="Priority" type="xs:int" />
        <xs:element minOccurs="0" name="Protocol" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="Service" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="Target" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="Weight" type="xs:int" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

**Port:** The port number of the service.

**Priority:** The priority of the target host. Lower values indicate greater priority.

**Protocol:** The transport protocol.

**Service:** The symbolic name of the desired service.

**Target:** The canonical name of the host providing the service.

**Weight:** A weighting factor for records with the same priority value.

### 2.2.4.9.5 ARecord

The **ARecord** type describes an A record in DNS.

```
<xs:complexType name="ARecord">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DnsRecord">
      <xs:sequence>
        <xs:element minOccurs="0" name="Address" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/System.Net" type="q1:IPAddress" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

**Address:** The address of the DNS record.

### 2.2.4.9.6 MXRecord

The **MXRecord** type describes a mail service (MX) record in DNS.

```
<xs:complexType name="MXRecord">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DnsRecord">
      <xs:sequence>
        <xs:element minOccurs="0" name="MailExchange" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="MailExchangePreference" type="xs:int" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

**MailExchange:** The name or address of the mail Exchange server.

**MailExchangePreference:** The preference for mail Exchange server.

### 2.2.4.9.7 TxtRecord

The **TxtRecord** type describes a text record in DNS.

```
<xs:complexType name="TxtRecord">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DnsRecord">
      <xs:sequence>
        <xs:element minOccurs="0" name="Text" nillable="true" type="xs:string" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

**Text:** The text for the record.

### 2.2.4.9.8 CNameRecord

The **CNameRecord** type describes a canonical name record in DNS.

```
<xs:complexType name="CNameRecord">
```

```
      <xs:complexContent mixed="false">
        <xs:extension base="tns:DnsRecord">
          <xs:sequence>
            <xs:element minOccurs="0" name="CanonicalName" nillable="true" type="xs:string" />
          </xs:sequence>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
```

**CanonicalName:** The canonical name for the record.

### 2.2.4.9.9 DomainState

The **DomainState** type describes the status of the domain.

```
    <xs:complexType name="DomainState">
      <xs:sequence>
        <xs:element minOccurs="0" name="ExpirationDate" type="xs:dateTime" />
        <xs:element minOccurs="0" name="Status" type="tns:DomainStatus" />
        <xs:element minOccurs="0" name="StatusDescription" nillable="true" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
```

**ExpirationDate:** The expiration date.

**Status:** The domain status.

**StatusDescription:** The description for domain status.

### 2.2.4.9.10    CertificateState

The **CertificateState** type describes the status of the certificate.

```
    <xs:complexType name="CertificateState">
      <xs:sequence>
        <xs:element minOccurs="0" name="Status" type="tns:CertificateStatus" />
        <xs:element minOccurs="0" name="StatusDescription" nillable="true" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
```

**Status:** The certificate status.

**StatusDescription:** The description of the certificate status.

### 2.2.4.9.11    DomainNameConfiguration

The **DomainNameConfiguration** type represents the currently active domain name configuration.

```
    <xs:complexType name="DomainNameConfiguration">
      <xs:sequence>
        <xs:element minOccurs="0" name="AccountName" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="CertificateRenewInProgress" type="xs:boolean" />
        <xs:element minOccurs="0" name="CertificateRenewStatus" type="tns:CertificateStatus" />
        <xs:element minOccurs="0" name="CertificateStatus" type="tns:CertificateStatus" />
        <xs:element minOccurs="0" name="CertificateStatusDescription" nillable="true"
    type="xs:string" />
        <xs:element minOccurs="0" name="AutoConfiguredAnywhereAccessHostName" type="tns:Idn" />
        <xs:element minOccurs="0" name="AutoConfiguredInternetDomainFullName" type="tns:Idn" />
        <xs:element minOccurs="0" name="DomainPrefix" type="tns:Idn" />
        <xs:element minOccurs="0" name="DomainStatus" type="tns:DomainStatus" />
```

```
            <xs:element minOccurs="0" name="DomainStatusDescription" nillable="true" type="xs:string"
/>
            <xs:element minOccurs="0" name="ExpirationDate" type="xs:dateTime" />
            <xs:element minOccurs="0" name="ManualModeEnabled" type="xs:boolean" />
            <xs:element minOccurs="0" name="Offering" nillable="true" type="tns:Offering" />
            <xs:element minOccurs="0" name="ProviderID" type="ser:guid" />
        </xs:sequence>
</xs:complexType>
```

**AccountName:** The account name used by the user to configure this offering.

**CertificateRenewInProgress:** TRUE, if a certificate renew is in progress; otherwise, FALSE.

**CertificateRenewStatus:** The status of the certificate renew task.

**CertificateStatus:** The certificate status.

**CertificateStatusDescription:** The description of the certificate status.

**AutoConfiguredAnywhereAccessHostName:** If the domain was automatically configured, this element contains the relative domain name of a host address, which can then be used to access the server from other network locations.

**AutoConfiguredInternetDomainFullName:** If the domain was automatically configured, this element contains the absolute domain name of the forward-lookup domain zone on the Internet.

**DomainPrefix:** The domain name prefix.

**DomainStatus:** The domain status.

**DomainStatusDescription:** The description for the domain status.

**ExpirationDate:** The expiration date.

**ManualModeEnabled:** TRUE, if manual mode is enabled; otherwise, FALSE.

**Offering:** The domain offering.

**ProviderID:** The provider ID.

### 2.2.4.9.12    Offering

The **Offering** type describes the domain name offering.

```
    <xs:complexType name="Offering">
      <xs:sequence>
        <xs:element minOccurs="0" name="AccountManagementUrl" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="AdditionalSites" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/System.Collections.ObjectModel"
type="q2:ReadOnlyCollectionOfanyURI" />
        <xs:element minOccurs="0" name="Description" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="EulaUrl" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="Icon" nillable="true" type="xs:base64Binary" />
        <xs:element minOccurs="0" name="MembershipServiceName" nillable="true" type="xs:string"
/>
        <xs:element minOccurs="0" name="MembershipServiceSignupUrl" nillable="true"
type="xs:anyURI" />
        <xs:element minOccurs="0" name="MoreInfoUrl" nillable="true" type="xs:anyURI" />
        <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="ProviderID" type="ser:guid" />
        <xs:element minOccurs="0" name="SupportedDomainType" type="tns:DomainType" />
        <xs:element minOccurs="0" name="TroubleshootingUrl" nillable="true" type="xs:anyURI" />
      </xs:sequence>
```

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

**AccountManagementUrl:** The account management URL.

**AdditionalSites:** A collection of additional sites.

**Description:** The description of the offering.

**EulaUrl:** The URL pointing to the EULA of the offering.

**Icon:** The icon of the offering.

**MembershipServiceName:** The membership service name.

**MembershipServiceSignupUrl:** The membership service signup URL.

**MoreInfoUrl:** The URL pointing to more information for the offering.

**Name:** The name of the offering.

**ProviderID:** The provider ID.

**SupportedDomainType:** A description of the supported domain type.

**TroubleshootingUrl:** A URL pointing to a troubleshooting webpage.

### 2.2.4.9.13    DynamicDNSInfo

The **DynamicDNSInfo** type describes the information of dynamic DNS.

```
    <xs:complexType name="DynamicDNSInfo">
      <xs:sequence>
        <xs:element minOccurs="0" name="FailedDNSUpdateAttempts" type="xs:int" />
        <xs:element minOccurs="0" name="LastSuccessfulDNSUpdate" type="xs:dateTime" />
        <xs:element minOccurs="0" name="LastUpdatedIPAddress" nillable="true"
  xmlns:q3="http://schemas.datacontract.org/2004/07/System.Net" type="q3:IPAddress" />
      </xs:sequence>
    </xs:complexType>
```

**FailedDNSUpdateAttempts:** Retrieves the number of unsuccessful DNS update attempts since the last successful update.

**LastSuccessfulDNSUpdate:** The time of the most recent successful DNS update.

**LastUpdatedIPAddress:** The most recently updated IP address by the dynamic DNS.

### 2.2.4.9.14    DomainProviderManagerSettings

The **DomainProviderManagerSettings** type describes settings for Domain Manager operation. At run-time, these settings are read-only. Administrators and/or tools can change these settings in the registry if needed.

```
    <xs:complexType name="DomainProviderManagerSettings">
      <xs:sequence>
        <xs:element minOccurs="0" name="BypassProxyForLocalAddresses" type="xs:boolean" />
        <xs:element minOccurs="0" name="DNSUpdateInterval" type="ser:duration" />
        <xs:element minOccurs="0" name="DomainStatusQueryInterval" type="ser:duration" />
        <xs:element minOccurs="0" name="Environment" type="tns:DomainProviderEnvironment" />
        <xs:element minOccurs="0" name="Proxy" nillable="true" type="xs:string" />
      </xs:sequence>
```

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

**BypassProxyForLocalAddresses:** TRUE, if the proxy for local addresses ~~should be~~is bypassed; otherwise, FALSE.

**DNSUpdateInterval:** The DNS update interval.

**DomainStatusQueryInterval:** The domain status query interval.

**Environment:** The environment in which the provider is running.

**Proxy:** The proxy setting.

### 2.2.4.10    Office 365 Integration Service

#### 2.2.4.10.1    O365ConfigurationException

The **O365ConfigurationException** type describes errors in Office 365 integration configurations.

```
<xs:complexType name="O365ConfigureException">
  <xs:complexContent mixed="false">
    <xs:extension xmlns:q1="http://schemas.datacontract.org/2004/07/System"
base="q1:Exception" />
  </xs:complexContent>
</xs:complexType>
```

### 2.2.4.11    Client Notification Service

#### 2.2.4.11.1    NotificationResult

The **NotificationResult** type describes the notification result in the client notification service.

```
    <xs:complexType name="NotificationResult">
      <xs:sequence>
        <xs:element minOccurs="0" name="clientName" nillable="true" type="xs:string" />
        <xs:element minOccurs="0" name="result" type="xs:boolean" />
      </xs:sequence>
    </xs:complexType>
```

#### 2.2.4.11.2    NotificationResults

The **NotificationResults** type describes notification results in the client notification service.

```
    <xs:complexType name="NotificationResults">
      <xs:sequence>
        <xs:list minOccurs="0" name="result" type="NotificationResult" />
      </xs:sequence>
    </xs:complexType>
```

### 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 |
|---|---|
| SourceLibrary (section 2.2.5.1.1) | The **SourceLibrary** type contains the options governing which source libraries the client file backup provider will back up. |
| ManagedStatus (section 2.2.5.1.2) | The **ManagedStatus** type describes the computer managed status. |
| AbortReason (section 2.2.5.2.1) | The **AbortReason** type describes why a client backup job is aborted. |
| PCBackupClientFaultReason (section 2.2.5.2.2) | The **PCBackupClientFaultReason** type describes the type of fault that occurred in the client backup provider. |
| PCBackupProviderState (section 2.2.5.2.3) | The **PCBackupProviderState** type describes the state of the current backup set operation. |
| PCBackupStatusOperation (section 2.2.5.2.4) | The **PCBackupStatusOperation** type describes the operation status of the backup service. |
| PCBackupVolumePhase (section 2.2.5.2.5) | The **PCBackupVolumePhase** type describes the state of the current backup volume operation. |
| CleanupStatus (section 2.2.5.2.6) | The **CleanupStatus** type describes the status of the cleanup job. |
| PCClientDataStatus (section 2.2.5.2.7) | The **PCClientDataStatus** type describes the status of backups for each client after repair. |
| ClientHealth (section 2.2.5.2.8) | The **ClientHealth** type describes the client backup health status. |
| ClientJobStatus (section 2.2.5.2.9) | The **ClientJobStatus** type describes the status of client backup job. |
| ConfigureBackupFaultReason (section 2.2.5.2.10) | The **ConfigureBackupFaultReason** type contains the reason for an error that occurred when configuring the computer backup. |
| DatabaseState (section 2.2.5.2.11) | The **DatabaseState** type describes the state of the computer backup database. |
| ExclusionReason (section 2.2.5.2.12) | The **ExclusionReason** type describes why a file is excluded. |
| FailureReason (section 2.2.5.2.13) | The **FailureReason** type describes the reason for a backup failure. |
| MountPhase (section 2.2.5.2.14) | The **MountPhase** type describes the different phases of mounting a backup. |
| PCBackupRepairState (section 2.2.5.2.15) | The **PCBackupRepairState** type describes the state of a backup repair operation. |
| RetentionState (section 2.2.5.2.16) | The **RetentionState** type describes backup jobs retention state. |
| StopReason (section 2.2.5.2.17) | The **StopReason** type describes why the computer backup service is stopped. |
| VolumeJobStatus (section 2.2.5.2.18) | The **VolumeJobStatus** type describes the configuration status of a volume. |
| VolumeLocation (section 2.2.5.2.19) | The **VolumeLocation** type indicates the location of a volume. |
| VolumeStatus (section 2.2.5.2.20) | The **VolumeStatus** type describes the configuration status of a volume. |
| PCBackupServerFaultReason (section 2.2.5.2.21) | The **PCBackupServerFaultReason** type indicates the type of fault that occurred in the client backup provider on the backup server. |
| TaskType (section 2.2.5.3.1) | The **TaskType** type represents type of tasks that the health provider |

| Simple Type | Description |
|---|---|
| | processes. |
| HealthStatus (section 2.2.5.3.2) | The **HealthStatus** type represents the status of a health alert. |
| ErrorCatalogType (section 2.2.5.4.1) | The **ErrorCatalogType** describes the error catalog of the Machine provider call. It is used as error code. |
| MachineStatus (section 2.2.5.4.2) | The **MachineStatus** type describes the current status of a machine. |
| WSSServerGroupType (section 2.2.5.5.1) | The **WSSServerGroupType** type describes the type of group to which the user belongs. |
| UserCustomExceptionCode (section 2.2.5.5.2) | The **UserCustomExceptionCode** type describes the definition of the error code for the user operation in the user provider. |
| RemoteAccessType (section 2.2.5.5.3) | The **RemoteAccessType** type describes the type of remote access. |
| AccessLevelType (section 2.2.5.5.4) | The **AccessLevelType** type describes the type of access level. |
| UserStatus (section 2.2.5.5.5) | The **UserStatus** type describes the user status. |
| CredentialType (section 2.2.5.6.1) | The **CredentialType** type describes the type of credentials that will be used for client authentication on the provider. |
| ConnectionSetting (section 2.2.5.6.2) | The **ConnectionSetting** type describes the connection setting to decide what type of sources are allowed to access the provider. |
| ImpersonationLevel (section 2.2.5.6.3) | The **ImpersonationLevel** type defines security impersonation levels. Security impersonation levels govern the degree to which a server process can act on behalf of a client process. |
| UpdateClassification (section 2.2.5.7.1) | The **UpdateClassification** type represents the different types of updates that an add-in can have. |
| OSEditions (section 2.2.5.7.2) | The **OSEditions** type describes information about the operating system that is used to limit which clients a file is installed to. |
| AddInError (section 2.2.5.7.3) | The **AddInError** type represents the error ID strings that can be reported back from the download provider. |
| FailureReason (section 2.2.5.8.1) | The **FailureReason** type represents the reason a failure occurred in the domain infrastructure service. |
| DomainStatus (section 2.2.5.8.2) | The **DomainStatus** type represents domain status from the point of view of the domain provider. |
| CertificateStatus (section 2.2.5.8.3) | The **CertificateStatus** type represents certificate status from the point of view of the domain provider. |
| DomainType (section 2.2.5.8.4) | The **DomainType** type describes the type of the domain supported. |
| DomainProviderEnvironment (section 2.2.5.8.5) | The **DomainProviderEnvironment** type describes the environment that the domain provider is running. |
| PasswordChangeStatus (section 2.2.5.9.2) | The **PasswordChangeStatus** type describes the status of a password change operation. |

## 2.2.5.1  Client File Backup Provider

## 2.2.5.1.1 SourceLibrary

The **SourceLibrary** type contains the options governing which source libraries the client file backup provider will back up.

```
<xs:simpleType name="SourceLibrary">
  <xs:restriction base="xs:string">
    <xs:enumeration value="DocumentsAndDesktop" />
    <xs:enumeration value="NoMediaLibraries" />
    <xs:enumeration value="All" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| DocumentsAndDesktop | The provider will back up the client's documents library and desktop items. |
| NoMediaLibraries | The provider will back up all files except media libraries (picture, video, music). |
| All | The provider will back up all files supported by the file history. |

### 2.2.5.1.2 ManagedStatus

The **ManagedStatus** type describes the computer managed status.

```
<xs:simpleType name="ManagedStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Managed" />
    <xs:enumeration value="NotManaged" />
    <xs:enumeration value="Unknown" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Managed | The file backup feature on the client is managed by the server. |
| NotManaged | The file backup feature on the client is not managed by the server. |
| Unknown | The managed status is unknown. |

### 2.2.5.2  Client Backup Provider

### 2.2.5.2.1 AbortReason

▪ The **AbortReason** type describes why a client backup job is aborted.

```
<xs:simpleType name="AbortReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None" />
    <xs:enumeration value="Authentication" />
    <xs:enumeration value="VersionMismatch" />
    <xs:enumeration value="Cancel" />
    <xs:enumeration value="ServerUnreachable" />
    <xs:enumeration value="ClientVssProblem" />
    <xs:enumeration value="ClientVssInsufficientSpace" />
```

```
            <xs:enumeration value="ClientIoError" />
            <xs:enumeration value="ClientException" />
            <xs:enumeration value="DBErrors" />
            <xs:enumeration value="ConnectionLost" />
            <xs:enumeration value="NotConfigured" />
            <xs:enumeration value="ServerFailure" />
            <xs:enumeration value="Unknown" />
            <xs:enumeration value="VolumeMissing" />
            <xs:enumeration value="CantOpenVolume" />
            <xs:enumeration value="BadFileSystem" />
            <xs:enumeration value="DBIsOffline" />
            <xs:enumeration value="EspCaptureFailed" />
        </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | No abort occurred; never send this. |
| Authentication | Authentication problem. |
| VersionMismatch | Mismatch between server side and client side protocol versions. |
| Cancel | Operation canceled by user. |
| ServerUnreachable | Server could not be contacted. |
| ClientVssProblem | VSS problem on client. |
| ClientVssInsufficientSpace | Insufficient space on client for a VSS snapshot. |
| ClientIoError | Disk I/O error on client. |
| ClientException | Unexpected exception on client. |
| DBErrors | Server database is corrupt. |
| ConnectionLost | Connection was lost after backup was started. |
| NotConfigured | Either the client or the volume is not configured for backup. |
| ServerFailure | Unexpected exception on server. |
| Unknown | Failed due to unknown reason. |
| VolumeMissing | Volume is currently missing. |
| CantOpenVolume | Volume could not be opened. |
| BadFileSystem | Volume's file system is not supported. |
| DBIsOffline | Backup database is currently offline. |
| EspCaptureFailed | Backup engine failed to capture ESP. |

## 2.2.5.2.2 PCBackupClientFaultReason

The **PCBackupClientFaultReason** type describes the type of fault that occurred in the client backup provider.

```
        <xs:simpleType name="PCBackupClientFaultReason">
```

```
      <xs:restriction base="xs:string">
        <xs:enumeration value="ServerNotReachable" />
        <xs:enumeration value="ClientNotConfigured" />
        <xs:enumeration value="Unknown" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| ServerNotReachable | ServerProvider is not reachable. |
| ClientNotConfigured | Client is not configured. |
| Unknown | Unknown error. |

### 2.2.5.2.3 PCBackupProviderState

The **PCBackupProviderState** type describes the state of the current backup set operation.

```
    <xs:simpleType name="PCBackupProviderState">
      <xs:restriction base="xs:string">
        <xs:enumeration value="Disconnected" />
        <xs:enumeration value="NotConfigured" />
        <xs:enumeration value="Connected" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Disconnected | Server is not available. |
| NotConfigured | Client is not configured for backup. |
| Connected | Server is connected. |

### 2.2.5.2.4 PCBackupStatusOperation

The **PCBackupStatusOperation** type describes the operation status of the backup service.

```
    <xs:simpleType name="PCBackupStatusOperation">
      <xs:restriction base="xs:string">
        <xs:enumeration value="None" />
        <xs:enumeration value="DeleteBackup" />
        <xs:enumeration value="Cleanup" />
        <xs:enumeration value="Backup" />
        <xs:enumeration value="PrepareRestore" />
        <xs:enumeration value="BasicChecker" />
        <xs:enumeration value="ComprehensiveChecker" />
        <xs:enumeration value="Suspend" />
        <xs:enumeration value="Repair" />
        <xs:enumeration value="Unknown" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | Backup server is idle. |
| DeleteBackup | Delete backups of a volume OR remove machine. |
| Cleanup | Database cleanup is in progress. |
| Backup | Backup operation is in progress. |
| PrepareRestore | Preparation phase of restore is in progress |
| BasicChecker | Basic consistency checker (BCC). |
| ComprehensiveChecker | Comprehensive consistency checker (CCC). |
| Suspend | A suspend operation (external). |
| Repair | Backup repair is in progress. |
| Unknown | Backup service is not reachable. |

### 2.2.5.2.5 PCBackupVolumePhase

The **PCBackupVolumePhase** type describes the state of the current backup volume operation.

```
<xs:simpleType name="PCBackupVolumePhase">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Idle" />
    <xs:enumeration value="ScanFileRecords" />
    <xs:enumeration value="ScanClusters" />
    <xs:enumeration value="Reorganize1" />
    <xs:enumeration value="SendClusters" />
    <xs:enumeration value="Reorganize2" />
    <xs:enumeration value="Complete" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Idle | No volume operation in progress. |
| ScanFileRecords | Phase 1: Determining which file records have changed. |
| ScanClusters | Phase 2: Determining which clusters have changed. |
| Reorganize1 | Phase 3: Reorganizing data on the server. |
| SendClusters | Phase 4: Sending data to the server. |
| Reorganize2 | Phase 5: Reorganizing data on the server. |
| Complete | Volume completed. |

### 2.2.5.2.6 CleanupStatus

The **CleanupStatus** type describes the status of the cleanup job.

```
<xs:simpleType name="CleanupStatus">
```

```
      <xs:restriction base="xs:string">
        <xs:enumeration value="None" />
        <xs:enumeration value="Running" />
        <xs:enumeration value="CancelRequested" />
        <xs:enumeration value="Completed" />
        <xs:enumeration value="Cancelled" />
        <xs:enumeration value="CleanupStatusPendingRestore" />
        <xs:enumeration value="CleanupStatusDBErrors" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | No cleanup attempted since the last server start. |
| Running | Cleanup is currently running. |
| CancelRequested | Cleanup is running; cancel has been requested. |
| Completed | Most recent cleanup completed successfully. |
| Cancelled | Most recent cleanup was canceled. |
| CleanupStatusPendingRestore | Cleanup could not run due to pending restores. |
| CleanupStatusDBErrors | Cleanup could not run due to database errors. |

## 2.2.5.2.7 PCClientDataStatus

The **PCClientDataStatus** type describes the status of backups for each client after repair.

```
    <xs:simpleType name="PCClientDataStatus">
      <xs:restriction base="xs:string">
        <xs:enumeration value="NoLoss" />
        <xs:enumeration value="PartialLoss" />
        <xs:enumeration value="FullLoss" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| NoLoss | All backups were preserved after repair. |
| PartialLoss | Some backups were lost after repair. |
| FullLoss | All backups were lost for this machine. |

## 2.2.5.2.8 ClientHealth

The **ClientHealth** type describes the client backup health status.

```
    <xs:simpleType name="ClientHealth">
      <xs:restriction base="xs:string">
        <xs:enumeration value="Unknown" />
        <xs:enumeration value="Normal" />
        <xs:enumeration value="Warning" />
```

```
          <xs:enumeration value="Error" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---------|-------------------------|
| Unknown | No information. |
| Normal | Backups on schedule. |
| Warning | Missed a few backups. |
| Error | Missed several backups. |

### 2.2.5.2.9 ClientJobStatus

The **ClientJobStatus** type describes the status of client backup job.

```
    <xs:simpleType name="ClientJobStatus">
      <xs:restriction base="xs:string">
        <xs:enumeration value="Complete" />
        <xs:enumeration value="Incomplete" />
        <xs:enumeration value="Failed" />
        <xs:enumeration value="InProgress" />
        <xs:enumeration value="InQueue" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|------------|------------------------------------------------------------------------------------------------------------------------------------|
| Complete | Job completed successfully. All configured volumes were backed up successfully. |
| Incomplete | Backup of at least one configured volume failed or skipped because volume was not available or broken because of data loss after backup. |
| Failed | Backup of every volume failed or every volume was skipped. |
| InProgress | Backup job is in progress. Backup of at least one volume is in progress or not started. |
| InQueue | Backup job is queued/waiting. |

### 2.2.5.2.10    ConfigureBackupFaultReason

The **ConfigureBackupFaultReason** type contains the reason for an error that occurred when configuring the computer backup.

```
    <xs:simpleType name="ConfigureBackupFaultReason">
      <xs:restriction base="xs:string">
        <xs:enumeration value="ConfigurationInProgress" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| ConfigurationInProgress | Another configuration is in progress. |

### 2.2.5.2.11 DatabaseState

The **DatabaseState** type describes the state of the computer backup database.

```
<xs:simpleType name="DatabaseState">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="Healthy" />
    <xs:enumeration value="DBHasErrors" />
    <xs:enumeration value="DBIsNew" />
    <xs:enumeration value="DBIsOld" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Database state is unknown. |
| Healthy | Database is online without any errors. |
| DBHasErrors | Database is online with errors. |
| DBIsNew | Database is offline because it is newer than code. |
| DBIsOld | Database is offline because it is older than code. |

### 2.2.5.2.12 ExclusionReason

The **ExclusionReason** type describes why a file is excluded.

```
<xs:simpleType name="ExclusionReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotExcluded" />
    <xs:enumeration value="ClientSideCache" />
    <xs:enumeration value="TemporaryFiles" />
    <xs:enumeration value="DigitalVideoRecorder" />
    <xs:enumeration value="PageFile" />
    <xs:enumeration value="RecycleBin" />
    <xs:enumeration value="HibernationFile" />
    <xs:enumeration value="MemoryDump" />
    <xs:enumeration value="ShadowVolumes" />
    <xs:enumeration value="Other" />
    <xs:enumeration value="Count" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| NotExcluded | File is not excluded. |
| ClientSideCache | File is excluded because it is a client-side cache file. |

| Value | Description |
|---|---|
| TemporaryFiles | File is excluded because it is a temporary file. |
| DigitalVideoRecorder | File is excluded because it is a digital video recorder file. |
| PageFile | File is excluded because it is a page file. |
| RecycleBin | File is excluded because it is in the recycle bin. |
| HibernationFile | File is excluded because it is a hibernation file. |
| MemoryDump | File is excluded because it is a memory dump. |
| ShadowVolumes | File is excluded because it is a shadow volume file. |
| Other | File is excluded because of some other reason. |
| Count | Count of items in this enumeration. |

### 2.2.5.2.13    FailureReason

The **FailureReason** type describes the reason for a backup failure.

```
<xs:simpleType name="FailureReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="NotAuthed" />
    <xs:enumeration value="InvalidSID" />
    <xs:enumeration value="ServerBusy" />
    <xs:enumeration value="MaxRestoreSessions" />
    <xs:enumeration value="DBErrors" />
    <xs:enumeration value="VolumeMissing" />
    <xs:enumeration value="BackupSetMissing" />
    <xs:enumeration value="BackupFailed" />
    <xs:enumeration value="RestoreVeriferFailed" />
    <xs:enumeration value="DBIsOffline" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Unknown failure reason. |
| NotAuthed | User is not authorized to perform this operation. |
| InvalidSID | Invalid client SID. |
| ServerBusy | Server is busy. |
| MaxRestoreSessions | Reached maximum number of simultaneous restore sessions. |
| DBErrors | Backup database has errors. |
| VolumeMissing | Requested volume GUID not in the volume configuration file. |
| BackupSetMissing | Requested backup set not in backup set file. |
| BackupFailed | Requested backup is marked as failed backup. |

| Value | Description |
|---|---|
| RestoreVeriferFailed | Restore verifier detected an error during preparation. |
| DBIsOffline | Backup database is currently offline. |

### 2.2.5.2.14   MountPhase

The **MountPhase** type describes the different phases of mounting a backup.

```
<xs:simpleType name="MountPhase">
  <xs:restriction base="xs:string">
    <xs:enumeration value="MountConnect" />
    <xs:enumeration value="MountAuthenticate" />
    <xs:enumeration value="MountLocateComputer" />
    <xs:enumeration value="MountLocateBackup" />
    <xs:enumeration value="MountLocateVolume" />
    <xs:enumeration value="MountMountVolume" />
    <xs:enumeration value="MountExploreVolume" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| MountConnect | Connect to the server. |
| MountAuthenticate | Authenticate the connection. |
| MountLocateComputer | Locate source computer. |
| MountLocateBackup | Locate source backup set. |
| MountLocateVolume | Locate source volume. |
| MountMountVolume | Actually mount the volume. |
| MountExploreVolume | Allow exploring the mounted volume. |

### 2.2.5.2.15   PCBackupRepairState

The **PCBackupRepairState** type describes the state of a backup repair operation.

```
<xs:simpleType name="PCBackupRepairState">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotStarted" />
    <xs:enumeration value="RepairNoErrors" />
    <xs:enumeration value="RepairDBIsNew" />
    <xs:enumeration value="RepairDBIsOld" />
    <xs:enumeration value="RepairInProgress" />
    <xs:enumeration value="RepairSucceeded" />
    <xs:enumeration value="RepairFailed" />
    <xs:enumeration value="RepairCanceled" />
    <xs:enumeration value="Unknown" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| NotStarted | Repair is starting. Not started yet. |
| RepairNoErrors | No database errors. Repair did not run. |
| RepairDBIsNew | Database is new. Repair did not run. |
| RepairDBIsOld | Database is old. Repair did not run. |
| RepairInProgress | Repair is running. |
| RepairSucceeded | Repair succeeded, for example, fixed all errors. |
| RepairFailed | Repair failed, for example, could not fix all errors. |
| RepairCanceled | Repair was canceled by user. |
| Unknown | Repair state is unknown. |

### 2.2.5.2.16    RetentionState

The **RetentionState** type describes backup jobs retention state.

```
<xs:simpleType name="RetentionState">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Delete" />
    <xs:enumeration value="Normal" />
    <xs:enumeration value="Locked" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Delete | Delete this job at next cleanup. |
| Normal | Delete this job when it is outside of retention policy. |
| Locked | Do not automatically delete this job. |

### 2.2.5.2.17    StopReason

The **StopReason** type describes why the computer backup service is stopped.

```
<xs:simpleType name="StopReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None" />
    <xs:enumeration value="DatabaseRestore" />
    <xs:enumeration value="DatabaseRepair" />
    <xs:enumeration value="DatabaseLocationChanged" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | This is not used. |
| DatabaseRestore | Service is stopped because the database is being restored. |
| DatabaseRepair | Service is stopped because the database is being repaired. |
| DatabaseLocationChanged | Service is stopped because the database location has changed. |

### 2.2.5.2.18    VolumeJobStatus

The **VolumeJobStatus** type describes the configuration status of a volume.

```
<xs:simpleType name="VolumeJobStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Complete" />
    <xs:enumeration value="Broken" />
    <xs:enumeration value="Failed" />
    <xs:enumeration value="Cancelled" />
    <xs:enumeration value="Skipped" />
    <xs:enumeration value="CheckPoint" />
    <xs:enumeration value="NotStarted" />
    <xs:enumeration value="InProgress" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Complete | Volume backup completed successfully. Volume can be used for restoration successfully. |
| Broken | Volume backup is broken due to data loss. Volume can be used for restoration but restore might fail. |
| Failed | Volume backup failed. Volume backup cannot be used for restoration. |
| Cancelled | Volume backup canceled. Volume backup cannot be used for restoration. |
| Skipped | Volume backup skipped. Volume backup cannot be used for restoration. |
| CheckPoint | Volume backup was completed but marked as a checkpoint. Volume backup cannot be used for restoration. |
| NotStarted | Volume backup has not started yet. Volume backup cannot be used for restoration. |
| InProgress | Volume backup is in progress. |

### 2.2.5.2.19    VolumeLocation

The **VolumeLocation** type indicates the location of a volume.

```
<xs:simpleType name="VolumeLocation">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Missing" />
    <xs:enumeration value="Internal" />
    <xs:enumeration value="External" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Missing | Volume is not attached to the client. |
| Internal | Volume is attached and not removable. |
| External | Volume is attached but removable. |

### 2.2.5.2.20    VolumeStatus

The **VolumeStatus** type describes the configuration status of a volume.

```
<xs:simpleType name="VolumeStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotConfigured" />
    <xs:enumeration value="Exclude" />
    <xs:enumeration value="Include" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| NotConfigured | User has not indicated whether to back up this volume. |
| Exclude | Exclude this volume from backups. |
| Include | Include this volume in backups. |

### 2.2.5.2.21    PCBackupServerFaultReason

The **PCBackupServerFaultReason** type indicates the type of fault that occurred in the client backup provider on the backup server.

```
<xs:simpleType name="PCBackupServerFaultReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ClientNotReachable" />
    <xs:enumeration value="ClientNotConfigured" />
    <xs:enumeration value="ServiceBusy" />
    <xs:enumeration value="ServiceNotReachable" />
    <xs:enumeration value="ClientNotFound" />
    <xs:enumeration value="ClientJobNotFound" />
    <xs:enumeration value="ClientVolumeNotFound" />
    <xs:enumeration value="RepairRunning" />
    <xs:enumeration value="DBHasErrors" />
    <xs:enumeration value="DBIsOffline" />
    <xs:enumeration value="ArgumentOutOfRange" />
    <xs:enumeration value="UnexpectedError" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| ClientNotReachable | Client cannot be reached. |

| Value | Description |
|---|---|
| ClientNotConfigured | Client is not configured for backup. |
| ServiceBusy | Backup service is busy. |
| ServiceNotReachable | Backup service cannot be reached. |
| ClientNotFound | Client was not found. |
| ClientJobNotFound | Client job was not found. |
| ClientVolumeNotFound | Client volume was not found. |
| RepairRunning | Repair operation is running. |
| DBHasErrors | Error occurred in the backup database. |
| DBIsOffline | Backup database is offline. |
| ArgumentOutOfRange | Invalid argument. |
| UnexpectedError | Unexpected error occurred. |

## 2.2.5.3  Health Provider

### 2.2.5.3.1 TaskType

The **TaskType** type represents type of tasks that the health provider processes.

```
<xs:simpleType name="TaskType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Raise" />
    <xs:enumeration value="Evaluate" />
    <xs:enumeration value="Clear" />
    <xs:enumeration value="Update" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Raise | Represents raising an alert via an API call. |
| Evaluate | Represents evaluating local alerts. |
| Clear | Clearing an alert via an API call. |
| Update | Updates an alert via an API call. |

### 2.2.5.3.2 HealthStatus

The **HealthStatus** type represents the status of a health alert.

```
<xs:simpleType name="HealthStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="OK" />
    <xs:enumeration value="Information" />
    <xs:enumeration value="Warning" />
```

```
        <xs:enumeration value="Critical" />
        <xs:enumeration value="Report" />
    </xs:restriction>
  </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| OK | The alert is not active. |
| Information | The alert is an information alert. |
| Warning | The alert is a warning alert. |
| Critical | The alert is a critical alert. |
| Report | The alert is a report containing additional information. |

## 2.2.5.4  Machine Identity Provider

### 2.2.5.4.1 ErrorCatalogType

The **ErrorCatalogType** describes the error catalog of the Machine provider call. It is used as error code.

```
<xs:simpleType name="ErrorCatalogType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Succeeded" />
    <xs:enumeration value="MachineExists" />
    <xs:enumeration value="NetworkError" />
    <xs:enumeration value="CAError" />
    <xs:enumeration value="InvalidUser" />
    <xs:enumeration value="InvalidOwnerOfMachine" />
    <xs:enumeration value="InvaildCert" />
    <xs:enumeration value="ProviderNotAvailable" />
    <xs:enumeration value="AccessDenied" />
    <xs:enumeration value="TooManyComputers" />
    <xs:enumeration value="OtherError" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Succeeded | The call succeeded. |
| MachineExists | Found the same Certificate in the Server, and the Enroll is not a forced enrollment. |
| NetworkError | An HTTPS error occurred. |
| CAError | Any error reported back from the certificate authority. |
| InvalidUser | The user credentials are incorrect. The detailed error code will be a native error code. |
| InvalidOwnerOfMachine | The machine object is not owned by the user. |
| InvaildCert | The certificate has expired. |

| Value | Description |
|---|---|
| ProviderNotAvailable | The dependent provider is not available. |
| AccessDenied | The current user does not have the required permissions. |
| TooManyComputers | Too many computers. |
| OtherError | All other errors. |

### 2.2.5.4.2 MachineStatus

The **MachineStatus** type describes the current status of a machine.

```
<xs:simpleType name="MachineStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Active" />
    <xs:enumeration value="Deactive" />
    <xs:enumeration value="ActiveWithCert" />
    <xs:enumeration value="DeactiveWithCert" />
    <xs:enumeration value="NotExists" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Active | The machine is active. |
| Deactive | The machine is disabled or inactive. |
| ActiveWithCert | The machine is active with a certificate. |
| DeactiveWithCert | The machine is disabled or inactive with a certificate. |
| NotExists | The machine does not exist. |

### 2.2.5.5  User Identity Provider

### 2.2.5.5.1 WSSSServerGroupType

The **WSSSServerGroupType** type describes the type of group to which the user belongs.

```
<xs:simpleType name="WSSSServerGroupType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="InvalidAccess" />
    <xs:enumeration value="RemoteAccess" />
    <xs:enumeration value="ShareAccess" />
    <xs:enumeration value="ComputerAccess" />
    <xs:enumeration value="MediaAccess" />
    <xs:enumeration value="AddInAccess" />
    <xs:enumeration value="DashboardAccess" />
    <xs:enumeration value="HomePageLinks" />
    <xs:enumeration value="NetworkAlertAccess" />
    <xs:enumeration value="VpnAccess" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| InvalidAccess | InvalidAccess Group. |
| RemoteAccess | Global group that defines whether the user can log into the Remote Web Access (RWA). |
| ShareAccess | A user in this group can see the same shares they would if the user were internal on the network. |
| ComputerAccess | A user in this group can connect to the computers the user could if they were internal on the network. |
| MediaAccess | A user in this group has access to media components of the RWA as well as WMP RME Vail. |
| AddInAccess | A user in this group can see web-enabled add-ins. |
| DashboardAccess | A user in this group gets a link to the server dashboard in their RWA. |
| HomePageLinks | A user in this group can see the administrator-defined links on the home page. |
| NetworkAlertAccess | Network Alert Administrators. The user in this group is the alert administrator. |
| VpnAccess | A user in this group can access via VPN. |

### 2.2.5.5.2 UserCustomExceptionCode

The **UserCustomExceptionCode** type describes the definition of the error code for the user operation in the user provider.

```
<xs:simpleType name="UserCustomExceptionCode">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Succeeded" />
    <xs:enumeration value="WinLogonFailed" />
    <xs:enumeration value="UserExists" />
    <xs:enumeration value="UserNotExist" />
    <xs:enumeration value="ProviderNotAvailable" />
    <xs:enumeration value="AccessDenied" />
    <xs:enumeration value="TooManyUsers" />
    <xs:enumeration value="CreateUserSharedFolderFailed" />
    <xs:enumeration value="Failed" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Succeeded | The user operation succeeded. |
| WinLogonFailed | Account does not exist, or the password of the account is wrong. |
| UserExists | User already exists. |
| UserNotExist | User does not exist. |
| ProviderNotAvailable | Dependent provider not available. |
| AccessDenied | Caller does not have the access required for this operation. |
| TooManyUsers | User number limit exceeds the policy per the Eula. |

| Value | Description |
|---|---|
| CreateUserSharedFolderFailed | Cannot create user folder. |
| Failed | General failure for any other reason. |

### 2.2.5.5.3 RemoteAccessType

The **RemoteAccessType** type describes the type of remote access.

```
<xs:simpleType name="RemoteAccessType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="Allowed" />
    <xs:enumeration value="NotAllowed" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Remote access type is unknown. |
| Allowed | Remote access is allowed. |
| NotAllowed | Remote access is not allowed. |

### 2.2.5.5.4 AccessLevelType

The **AccessLevelType** type describes the type of access level.

```
<xs:simpleType name="AccessLevelType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="Guest" />
    <xs:enumeration value="User" />
    <xs:enumeration value="Administrator" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Access level type is unknown. |
| Guest | Guest account. |
| User | Standard user account. |
| Administrator | Administrator account. |

### 2.2.5.5.5 UserStatus

The **UserStatus** type describes the user status.

```
<xs:simpleType name="UserStatus">
```

```
      <xs:restriction base="xs:string">
        <xs:enumeration value="Disabled" />
        <xs:enumeration value="Enabled" />
        <xs:enumeration value="Error" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Disabled | It is disabled. |
| Enabled | It is enabled. |
| Error | User creation encountered an error. |

### 2.2.5.6  Provider Registry Service

### 2.2.5.6.1 CredentialType

The **CredentialType** type describes the type of credentials that will be used for client authentication on the provider.

```
    <xs:simpleType name="CredentialType">
      <xs:restriction base="xs:string">
        <xs:enumeration value="User" />
        <xs:enumeration value="Machine" />
        <xs:enumeration value="None" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| User | Client credential type will be tied to the user.<8> |
| Machine | Client credential type will be tied to the machine.<9> |
| None | No client credential will be used. |

### 2.2.5.6.2 ConnectionSetting

The **ConnectionSetting** type describes the connection setting to decide what type of sources are allowed to access the provider.

```
    <xs:simpleType name="ConnectionSetting">
      <xs:restriction base="xs:string">
        <xs:enumeration value="AllowLocalAccessOnly" />
        <xs:enumeration value="AllowRemoteAccess" />
      </xs:restriction>
    </xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| AllowLocalAccessOnly | Allows the provider to accept incoming connections based on whether they are from the local host computer or the remote connection. |
| AllowRemoteAccess | Allows the provider to accept incoming connections from the local host and to have remote access to the host computer. |

## 2.2.5.6.3 ImpersonationLevel

The **ImpersonationLevel** type defines security impersonation levels. Security impersonation levels govern the degree to which a server process can act on behalf of a client process.

```
<xs:simpleType name="ImpersonationLevel">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Identification" />
    <xs:enumeration value="Impersonation" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Identification | The server process can obtain information about the client, such as security identifiers and privileges, but it cannot impersonate the client. This is useful for servers that export their own objects, such as database products that export tables and views. Using the retrieved client-security information, the server can make access-validation decisions without being able to use other services that are using the client's security context. |
| Impersonation | The server process can impersonate the client's security context on its local system, but not on remote systems. |

## 2.2.5.7 Addin Management

## 2.2.5.7.1 UpdateClassification

The **UpdateClassification** type represents the different types of updates that an add-in can have.

```
<xs:simpleType name="UpdateClassification">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="CriticalUpate" />
    <xs:enumeration value="ImportantUpdate" />
    <xs:enumeration value="Update" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Unknown is not a valid value to set using AddInManager.NewAddInVersionAvailable. |
| CriticalUpate | The update is critical. Note the idiosyncratic spelling of the value string. |
| ImportantUpdate | The update is important. |

| Value | Description |
|-------|-------------|
| Update | The update is less important. |

## 2.2.5.7.2 OSEditions

The **OSEditions** type describes information about the operating system that is used to limit which clients a file is installed to.

```
<xs:simpleType name="OSEditions">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">0</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="NotSet">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">0</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Home">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">1</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Pro">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">2</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Starter">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">4</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="MediaCenter">
        <xs:annotation>
            <xs:appinfo>
            <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">8</EnumerationValue>
            </xs:appinfo>
        </xs:annotation>
        </xs:enumeration>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | The filter is not based on an operating system (OS) edition. |
| NotSet | Compatible with published SDK. |
| Home | Home OS SKUs. |
| Pro | Pro OS SKUs. |
| Starter | Starter OS SKUs. |
| MediaCenter | OS editions that contain Media Center. |

### 2.2.5.7.3 AddInError

The **AddInError** type represents the error ID strings that can be reported back from the download provider.

```
<xs:simpleType name="AddInError">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None" />
    <xs:enumeration value="OperationCancelled" />
    <xs:enumeration value="UnknownException" />
    <xs:enumeration value="FileNotInPackage" />
    <xs:enumeration value="InvalidSignature" />
    <xs:enumeration value="DownloadError" />
    <xs:enumeration value="PackageExtraction" />
    <xs:enumeration value="InstallError" />
    <xs:enumeration value="UpgradeError" />
    <xs:enumeration value="UnableToQueryServer" />
    <xs:enumeration value="InvalidDownloadRequest" />
    <xs:enumeration value="UninstallError" />
    <xs:enumeration value="AuthorizationError" />
    <xs:enumeration value="ProviderConnectionError" />
    <xs:enumeration value="PackageFileCopyError" />
    <xs:enumeration value="SettingsProviderError" />
    <xs:enumeration value="InvalidData" />
    <xs:enumeration value="UnknownAddIn" />
    <xs:enumeration value="OperationInProgress" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | No error. |
| OperationCancelled | The asynchronous operation was successfully canceled. |
| UnknownException | An unknown error has occurred. |
| FileNotInPackage | The add-in package does not include an expected file. |
| InvalidSignature | The add-in package either does not have a signature when it is needed, or the signature is invalid. |
| DownloadError | An error occurred while downloading a file. |
| PackageExtraction | An error occurred while extracting files from the add-in package. |

| Value | Description |
|---|---|
| InstallError | An error occurred while installing an MSI. |
| UpgradeError | An error occurred while upgrading an MSI. |
| UnableToQueryServer | Unable to query server. |
| InvalidDownloadRequest | The download request was invalid (for example, an attempt to download using neither HTTP nor a signed package). |
| UninstallError | An error occurred while uninstalling an MSI. |
| AuthorizationError | Access denied. |
| ProviderConnectionError | An error occurred while connecting to the provider. |
| PackageFileCopyError | Could not copy add-in package. |
| SettingsProviderError | An error occurred while contacting the settings provider; changes will not be persisted. |
| InvalidData | The provided data is invalid; no changes have been made. |
| UnknownAddIn | The given add-in ID is not known; no action was taken. |
| OperationInProgress | An install/uninstall operation is already in progress. |

## 2.2.5.8  Domain Infrastructure Service

### 2.2.5.8.1 FailureReason

The **FailureReason** type represents the reason a failure occurred in the domain infrastructure service.

```
<xs:simpleType name="FailureReason">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="InvalidCredentials" />
    <xs:enumeration value="InvalidDomainName" />
    <xs:enumeration value="QuotaExceeded" />
    <xs:enumeration value="PurchaseNotStarted" />
    <xs:enumeration value="PurchaseFailed" />
    <xs:enumeration value="PendingCustomerInterventionRequired" />
    <xs:enumeration value="InRenewalCustomerInterventionRequired" />
    <xs:enumeration value="RenewalFailed" />
    <xs:enumeration value="RemoteProviderFailure" />
    <xs:enumeration value="CommunicationFailure" />
    <xs:enumeration value="LocalProviderFailure" />
    <xs:enumeration value="CertificateInstallFailure" />
    <xs:enumeration value="CertificateFileBroken" />
    <xs:enumeration value="CertificateNotTrusted" />
    <xs:enumeration value="CertificateBebeforeValidPeriod" />
    <xs:enumeration value="CertificateNoPrivateKey" />
    <xs:enumeration value="CertificateFileBrokenOrPasswordWrong" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Unknown failure. |

| Value | Description |
|---|---|
| InvalidCredentials | User credentials are incorrect. |
| InvalidDomainName | Domain name is not valid. |
| QuotaExceeded | Domain names under a user account exceeded the quota. |
| PurchaseNotStarted | User has not correctly initiated the purchase process. |
| PurchaseFailed | An error occurred when purchasing the domain name. |
| PendingCustomerInterventionRequired | Domain setup in progress; customer intervention needed. |
| InRenewalCustomerInterventionRequired | Domain renewal in progress; customer intervention needed. |
| RenewalFailed | Domain renewal failed. |
| RemoteProviderFailure | Remote provider failure. |
| CommunicationFailure | Communication failure. |
| LocalProviderFailure | Local provider failure. |
| CertificateInstallFailure | An error occurred when installing the certificate. |
| CertificateFileBroken | Certificate file is broken. |
| CertificateNotTrusted | Certificate is not trusted. |
| CertificateBebeforeValidPeriod | The current time precedes the valid certificate period. |
| CertificateNoPrivateKey | Certificate has no private key. |
| CertificateFileBrokenOrPasswordWrong | Certificate file is broken or the provider password is wrong. |

## 2.2.5.8.2 DomainStatus

The **DomainStatus** type represents domain status from the point of view of the domain provider.

```
<xs:simpleType name="DomainStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="NotConfigured">
      <xs:annotation>
        <xs:appinfo>
          <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">0</EnumerationValue>
        </xs:appinfo>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="Ready">
      <xs:annotation>
        <xs:appinfo>
          <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">1</EnumerationValue>
        </xs:appinfo>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="Pending">
      <xs:annotation>
        <xs:appinfo>
          <EnumerationValue
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">2</EnumerationValue>
        </xs:appinfo>
```

```
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="PendingCustomerInterventionRequired">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">3</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="Failed">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">4</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="PurchaseFailed">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">5</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="DomainNotFound">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">6</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="InRenewal">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">7</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="InRenewalCustomerInterventionRequired">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">8</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
        <xs:enumeration value="RenewalFailed">
          <xs:annotation>
            <xs:appinfo>
              <EnumerationValue
  xmlns="http://schemas.microsoft.com/2003/10/Serialization/">9</EnumerationValue>
            </xs:appinfo>
          </xs:annotation>
        </xs:enumeration>
    </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Domain status is unknown. |

| Value | Description |
|---|---|
| NotConfigured | Domain is not configured |
| Ready | Domain name is ready. DNS data ~~may~~might still be propagating. |
| Pending | Domain setup in progress; no customer intervention needed. |
| PendingCustomerInterventionRequired | Domain setup in progress; customer intervention needed. |
| Failed | Domain setup failed without a specific reason. |
| PurchaseFailed | Purchase of domain failed. |
| DomainNotFound | Domain is not found for this credential. This usually means that the purchase has not started. |
| InRenewal | Domain renewal in progress; no customer intervention needed. |
| InRenewalCustomerInterventionRequired | Domain renewal in progress; customer intervention needed. |
| RenewalFailed | Domain renewal failed. |

### 2.2.5.8.3 CertificateStatus

The **CertificateStatus** type represents certificate status from the point of view of the domain provider.

```
<xs:simpleType name="CertificateStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="Ready" />
    <xs:enumeration value="Failed" />
    <xs:enumeration value="CertificateRequestNotSubmitted" />
    <xs:enumeration value="Pending" />
    <xs:enumeration value="PendingCustomerInterventionRequired" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Certificate status is unknown. |
| Ready | Certificate has been successfully provisioned or renewed and is ready for downloading. |
| Failed | Certificate request failed. |
| CertificateRequestNotSubmitted | Certificate request has not been submitted. |
| Pending | Trusted certificate issuance pending; no customer intervention needed. |
| PendingCustomerInterventionRequired | Trusted certificate issuance pending; customer intervention needed. |

### 2.2.5.8.4 DomainType

The **DomainType** type describes the type of the domain supported.

```
<xs:simpleType name="DomainType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None" />
    <xs:enumeration value="Home" />
    <xs:enumeration value="Business" />
    <xs:enumeration value="All" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| None | Domain is not supported. |
| Home | Domain is for a home network. |
| Business | Domain is for a business network. |
| All | Domain is for any network. |

### 2.2.5.8.5 DomainProviderEnvironment

The **DomainProviderEnvironment** type describes the environment that the domain provider is running.

```
<xs:simpleType name="DomainProviderEnvironment">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Testing" />
    <xs:enumeration value="Production" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Testing | Domain provider is running in a testing environment. |
| Production | Domain provider is running in a production environment. |

### 2.2.5.9  AzureAD and HostedEmail Integration Service

AzureAD and HostedEmail integration service SHOULD<10> be implemented.

### 2.2.5.9.1 PasswordChangeType

The PasswordChangeType type describes the type of a password change operation.

```
<xs:simpleType name="PasswordChangeType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="AzureAD" />
    <xs:enumeration value="HostedEmail" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| AzureAD | The password change is for AzureAD. |
| HostedEmail | The password change is for HostedEmail. |

## 2.2.5.9.2 PasswordChangeStatus

The **PasswordChangeStatus** type describes the status of a password change operation.

```xml
<xs:simpleType name="PasswordChangeStatus">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Unknown" />
    <xs:enumeration value="Required" />
    <xs:enumeration value="UnSubmitted" />
    <xs:enumeration value="New" />
    <xs:enumeration value="InProcess" />
    <xs:enumeration value="Success" />
    <xs:enumeration value="WaitForRetry" />
    <xs:enumeration value="Failed" />
    <xs:enumeration value="Fixed" />
    <xs:enumeration value="InvalidPassword" />
  </xs:restriction>
</xs:simpleType>
```

The following table describes the values for this type.

| Value | Description |
|---|---|
| Unknown | Password change status is unknown. |
| Required | User is required to change the password, but it is not changed yet. |
| UnSubmitted | Password has not been submitted to the password sync receiver. |
| New | Password has been submitted to the password sync manager as a new event. |
| InProcess | Password change event needs to process. |
| Success | Succeeded in syncing with online. |
| WaitForRetry | Process failed within retry time, pending next retry. |
| Failed | Retry exceeded retry time. |
| Fixed | Marked fixed by end-users for failed data. |
| InvalidPassword | The password is invalid for Azure Active Directory.<11> |

# 3 Protocol Details

## 3.1 Client File Backup Provider Service Contract Details

### 3.1.1 Abstract Data Model

None.

### 3.1.2 Timers

None.

### 3.1.3 Initialization

None.

### 3.1.4 Higher-Layer Triggered Events

None.

### 3.1.5 Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IUserConfigProvider.GetUserSettingAsync | An asynchronous operation to obtain user settings. |
| IUserConfigProvider.GetClientManagedSettingAsync | An asynchronous operation to determine whether a client is managed. |
| IUserConfigProvider.GetUserSettingCompleted | The callback for the IUserConfigProvider.GetUserSettingAsync operation contract. |
| IUserConfigProvider.GetClientManagedCompleted | The callback for the IUserConfigProvider.GetClientManagedSettingAsync operation contract. |

#### 3.1.5.1 IUserConfigProvider.GetUserSettingAsync

This operation submits an asynchronous request to obtain user settings.

```
<wsdl:operation name="GetUserSettingAsync">
  <wsdl:input wsam:Action="http://tempuri.org/IUserConfigProvider/GetUserSettingAsync"
message="tns:IUserConfigProvider_GetUserSettingAsync_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IUserConfigProvider_GetUserSettingAsync_InputMessage** request message, the protocol server retrieves the specified user settings and returns them through the **IUserConfigProvider.GetUserSettingCompleted** callback.

#### 3.1.5.1.1 Messages

#### 3.1.5.1.1.1 IUserConfigProvider_GetUserSettingAsync_InputMessage

This is the request for the **GetUserSettingAsync** operation.

```
<wsdl:message name="IUserConfigProvider_GetUserSettingAsync_InputMessage">
  <wsdl:part name="parameters" element="tns:GetUserSettingAsync" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IUserConfigProvider/GetUserSettingAsyncResponse
```

The body of the SOAP message MUST contain the **GetUserSettingAsync** element.

### 3.1.5.1.2 Elements

### 3.1.5.1.2.1   GetUserSettingAsync

This element specifies input values for the **GetUserSettingAsync** operation.

```
<xs:element name="GetUserSettingAsync">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.1.5.2  IUserConfigProvider.GetClientManagedSettingAsync

This operation determines whether the client is managed.

```
<wsdl:operation name="GetClientManagedSettingAsync">
  <wsdl:input
wsam:Action="http://tempuri.org/IUserConfigProvider/GetClientManagedSettingAsync"
message="tns:IUserConfigProvider_GetClientManagedSettingAsync_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IUserConfigProvider_GetClientManagedSettingAsync_InputMessage** request message, the protocol server retrieves the specified user settings and returns them through the **IUserConfigProvider.GetClientManagedCompleted** callback.

### 3.1.5.2.1 Messages

### 3.1.5.2.1.1   IUserConfigProvider_GetClientManagedSettingAsync_InputMessage

This is the request for the **GetClientManagedSettingAsync** operation.

```
<wsdl:message name="IUserConfigProvider_GetClientManagedSettingAsync_InputMessage">
  <wsdl:part name="parameters" element="tns:GetClientManagedSettingAsync" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IUserConfigProvider/GetClientManagedSettingAsync
```

The body of the SOAP message MUST contain the **GetClientManagedSettingAsync** element.

### 3.1.5.2.2 Elements

#### 3.1.5.2.2.1 GetClientManagedSettingAsync

This element specifies input values for the **GetClientManagedSettingAsync** operation.

```
<xs:element name="GetClientManagedSettingAsync">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="machineName" nillable="true" type="xs:string"
        />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.1.5.3 IUserConfigProviderCallback.GetUserSettingCompleted

This operation is the callback associated with the **IUserConfigProvider.GetUserSettingAsync** operation.

```
<wsdl:operation name="GetUserSettingCompleted">
  <wsdl:output wsam:Action="http://tempuri.org/IUserConfigProvider/GetUserSettingCompleted"
    message="tns:IUserConfigProvider_GetUserSettingCompleted_OutputCallbackMessage"
  />
</wsdl:operation>
```

### 3.1.5.3.1 Messages

#### 3.1.5.3.1.1 IUserConfigProvider_GetUserSettingCompleted_OutputCallbackMessage

This is the request for the **GetUserSettingCompleted** operation.

```
<wsdl:message name="IUserConfigProviderCallback_GetUserSettingCompleted_InputMessage">
  <wsdl:part name="parameters" element="tns:GetUserSettingCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IUserConfigProviderCallback/GetUserSettingCompleted
```

The body of the SOAP message MUST contain the **GetUserSettingCompleted** element.

### 3.1.5.3.2 Elements

#### 3.1.5.3.2.1 GetUserSettingCompleted

This element specifies input values for the **GetUserSettingCompleted** operation.

```
<xs:element name="GetUserSettingCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="exception" nillable="true"
type="ClientFileBackupException" />
```

```
          <xs:element minOccurs="0" name="configSet" type="ConfigurationSet" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
```

## 3.1.5.4 IUserConfigProvider.GetClientManagedCompleted

This operation is the callback associated with the
**IUserConfigProvider.GetClientManagedSettingAsync** operation.

```
<wsdl:operation name="GetClientManagedCompleted">
  <wsdl:output wsam:Action="http://tempuri.org/IUserConfigProvider/GetClientManagedCompleted"
message="tns:IUserConfigProvider_GetClientManagedCompleted_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.1.5.4.1 Messages

#### 3.1.5.4.1.1    IUserConfigProvider_GetClientManagedCompleted_OutputCallbackMessage

This operation is the request for the **GetClientManagedCompleted** operation.

```
<wsdl:message name="IUserConfigProvider_GetClientManagedCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetClientManagedCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IUserConfigProvider/GetClientManagedCompleted
```

The body of the SOAP message MUST contain the **GetClientManagedCompleted** element.

### 3.1.5.4.2 Elements

#### 3.1.5.4.2.1    GetClientManagedCompleted

This element specifies input values for the **GetClientManagedCompleted** operation.

```
<xs:element name="GetClientManagedCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="exception" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.FileBackup" type="q3:ClientFileBackupException" />
      <xs:element minOccurs="0" name="isManaged"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.FileBackup" type="q4:ManagedStatus" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.1.6  Timer Events

None.

### 3.1.7  Other Local Events

None.

## 3.2    Client Backup Provider Service Contract Details

### 3.2.1   Abstract Data Model

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

### 3.2.1.1  Backup Server

The backup server maintains the following data elements.

### 3.2.1.1.1 List of Backup Client Configurations

The server maintains a list of backup configurations for each client.

### 3.2.1.1.2 List of Backup Status

The server maintains a list of backup statuses for each client. The protocol allows the backup client to request to change the status.

The following state machine diagram illustrates the relationship between backup status and series of backup requests.
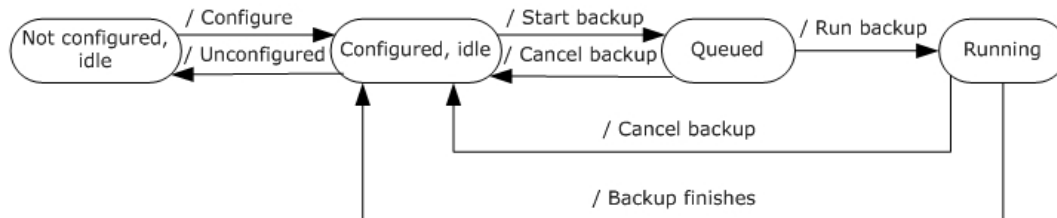


**Figure 2: Backup status machine diagram**

### 3.2.1.1.3 List of Backup Volume Configurations for Each Client

The server maintains a list of backup configurations for each volume of each client.

### 3.2.1.1.4 List of Backup Jobs

The server maintains a list of backup jobs, including jobs that are complete but not cleaned up, the job that is currently running, and jobs that are scheduled in the job queue.

### 3.2.1.1.5 Backup Window and Retention Policy

Specifies the start time and the end time of the backup window in which automatic backups are scheduled.

### 3.2.1.1.6 Retention Policy

Specifies the retention policy for old backups.

### 3.2.1.2 Backup Client

The backup clients maintain the following data elements:

- Priority: The priority of backup jobs for the client. This protocol only gets this value over the wire. This value can only be changed locally on the client by adding or changing a registry.

- Automatic backup schedule: The backup schedule of automatic backup jobs for the client. This is calculated from the backup window on the server. Each client has a different backup schedule; this prevents two clients from starting the automatic backup process at the same time. The schedule can be saved in the task scheduler.<12>

### 3.2.2 Timers

None.

### 3.2.3 Initialization

None.

### 3.2.4 Higher-Layer Triggered Events

None.

### 3.2.5 Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IPCBackupServerRegister.RegisterBackupClient | Registers the client's callback interface. Also returns client configuration from the server when called for the first time. For example, before the client sends any volume configuration information. |
| IPCBackupServerRegister.KeepAlive | Keeps the channel alive. |
| IPCBackupServerRegister.VolumeCalculatorProgress | Notifies the server of changes in progress on the volume tree calculation. |
| IPCBackupServerRegister.GetClientPriority | Returns the priority of the client. |
| IPCBackupServerRegister.OnBatteryPower | Returns whether the client is running on battery power. |
| IPCBackupServerRegister.StartBackup | Throws a fault of type invalid operation if the backup job is not in a stopped or started state. If the backup is already started, the operation has no effect and returns without any faults. |
| IPCBackupServerRegister.CancelBackup | Cancels the backup job without regard for its state. Does nothing if it is already idle. |
| IPCBackupServerRegistert.UpdateBackupConfiguration | Updates the backup configuration on the client whenever it changes. |
| IPCBackupServerRegister.SendBackupProgress | Updates the progress percentage of a running backup operation. |

| OperationContract | Description |
|---|---|
| IPCBackupServerRegister.GetFolderChildren | Gets folder children. Called after **CalculateVolume** finishes. |
| IPCBackupServerRegister.CalculateVolumeAsync | Calculates the folder tree for the volume. |
| IPCBackupServerRegister.CleanFolderSizeCalculation | Called by the server to clean the folder size calculation. Also called internally when the connection with the server breaks. |
| IPCBackupServerRegister.GetConnectedVolumes | Called by the server to get the list of volumes that are connected to the client. Some could be configured while others are not. |
| IPCBackupServerRegister.RescanVolumesAsync | Forces the client to send the volume configuration to the server. |
| IPCBackupServerProvider.Register | Registers the caller with the PC backup server provider. |
| IPCBackupServerProvider.GetClients | Operation to get list of all clients. |
| IPCBackupServerProvider.GetClient | Operation to get a specific client. |
| IPCBackupServerProvider.GetClientJobs | Returns the list of up to 100 client jobs from index. |
| IPCBackupServerProvider.GetVolumeJobs | Operation to get list of all volume jobs. |
| IPCBackupServerProvider.GetClientVolumes | Operation to get list of all client volumes. |
| IPCBackupServerProvider.SetClientJobDescription | Operation to set client job description. |
| IPCBackupServerProvider.SetClientJobRetentionState | Operation to set client job retention state. |
| IPCBackupServerProvider.SetClientVolumeStatus | Operation to set client volume status. |
| IPCBackupServerProvider.SetExcludedFolders | Operation to set excluded folders. |
| IPCBackupServerProvider.EnableBackups | Operation to enable backups on a client. |
| IPCBackupServerProvider.DisableBackups | Operation to disable backups on a client. |
| IPCBackupServerProvider.GetBackupConfiguration | Operation to get system wide backup configuration: backup windows, cleanup schedule etc. |
| IPCBackupServerProvider.GetDefaultBackupConfiguration | Operation to get default backup configuration: backup windows, cleanup schedule etc. |
| IPCBackupServerProvider.GetBackupOperationStatus | Operation to get backup operation status. |
| IPCBackupServerProvider.GetServiceStatus | Operation to get backup service status. |
| IPCBackupServerProvider.GetWaitingOperations | Operation to get waiting operations. |
| IPCBackupServerProvider.SetBackupConfiguration | Operation to set system wide backup configurations. |
| IPCBackupServerProvider.GetUsedBytes | Get total number of used bytes for all backups. |
| IPCBackupServerProvider.StartCleanup | Operation to start cleanup operation. |
| IPCBackupServerProvider.CancelCleanup | Operation to cancel cleanup operation. |

| OperationContract | Description |
|---|---|
| IPCBackupServerProvider.StartRepair | Operation to start repair operation. |
| IPCBackupServerProvider.CancelRepair | Operation to cancel repair operation. |
| IPCBackupServerProvider.GetLastRepairResult | Get last repair result. |
| IPCBackupServerProvider.StartConsistencyChecker | Operation to start consistency checker operation. |
| IPCBackupServerProvider.CancelConsistencyChecker | Operation to cancel consistency checker operation. |
| IPCBackupServerProvider.Remove | Operation to remove specific client. |
| IPCBackupServerProvider.CalculateVolumeAsync | Starts to calculate the specific volume. |
| IPCBackupServerProvider.GetFolderChildren | Get children of the specified folder. |
| IPCBackupServerProvider.GetConnectedVolumes | Get connected volumes of the specified client. |
| IPCBackupServerProvider.CleanFolderSizeCalculation | Cleanup folder size calculation. |
| IPCBackupServerProvider.StartBackup | Operation to start backup on a particular client. |
| IPCBackupServerProvider.CancelBackup | Operation to cancel backup on a particular client. |
| IPCBackupServerProvider.StartBackupService | Operation to start backup service. |
| IPCBackupServerProvider.StopBackupService | Operation to stop backup service. |
| IPCBackupServerProvider.NotifyBackupOperationStatusChange | Callback operation to notify caller of backup operation status change. |
| IPCBackupServerProvider.VolumeCalculatorProgress | Callback operation to notify caller of volume calculator progress change. |
| IPCBackupServerProvider.NotifyServiceStatusChange | Callback operation to notify caller of service status change. |

### 3.2.5.1 IPCBackupServerRegister.RegisterBackupClient

Registers the client's callback interface. Also returns client configuration from the server when called for the first time. For example, before the client sends any volume configuration information.

```
<wsdl:operation name="RegisterBackupClient">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerRegister/RegisterBackupClient"
message="tns:IPCBackupServerRegister_RegisterBackupClient_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerRegister/RegisterBackupClientResponse"
message="tns:IPCBackupServerRegister_RegisterBackupClient_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/RegisterBackupClientInvalidOperationE
xceptionFault" name="InvalidOperationExceptionFault"
message="tns:IPCBackupServerRegister_RegisterBackupClient_InvalidOperationExceptionFault_Faul
tMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_RegisterBackupClient_InputMessage** request message, the protocol server registers the callback interface. Upon successful completion of the

action, the server MUST respond with**IPCBackupServerRegister_RegisterBackupClient_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client, as specified in section 2.2.2.1.

### 3.2.5.1.1 Messages

#### 3.2.5.1.1.1 IPCBackupServerRegister_RegisterBackupClient_InputMessage

This is the request for the **RegisterBackupClient** operation.

```
<wsdl:message name="IPCBackupServerRegister_RegisterBackupClient_InputMessage">
  <wsdl:part name="parameters" element="tns:RegisterBackupClient" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/RegisterBackupClient
```

The body of the SOAP message MUST contain the **RegisterBackupClient** element.

#### 3.2.5.1.1.2 IPCBackupServerRegister_RegisterBackupClient_OutputMessage

This is the response for the **RegisterBackupClient** operation.

```
<wsdl:message name="IPCBackupServerRegister_RegisterBackupClient_OutputMessage">
  <wsdl:part name="parameters" element="tns:RegisterBackupClientResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/RegisterBackupClientResponse
```

The body of the SOAP message MUST contain the **RegisterBackupClientResponse** element.

### 3.2.5.1.2 Elements

#### 3.2.5.1.2.1 RegisterBackupClient

This element specifies input values for the **RegisterBackupClient** operation.

```
<xs:element name="RegisterBackupClient">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.2.5.1.2.2 RegisterBackupClientResponse

This element specifies output values for the **RegisterBackupClient** operation.

```
<xs:element name="RegisterBackupClientResponse">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="RegisterBackupClientResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q1:ClientConfig" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.2.5.2  IPCBackupServerRegister.KeepAlive

This operation only keeps the channel alive.

```
<wsdl:operation name="KeepAlive">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerRegister/KeepAlive"
message="tns:IPCBackupServerRegister_KeepAlive_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/KeepAliveResponse"
message="tns:IPCBackupServerRegister_KeepAlive_OutputMessage" />
</wsdl:operation>
```

Upon receiving **IPCBackupServerRegister_KeepAlive_InputMessage** request message, the
protocol server gets the priority of the client. Upon successful completion of the action, the server
MUST respond with **IPCBackupServerRegister_KeepAlive_OutputMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.2.1 Messages

#### 3.2.5.2.1.1    IPCBackupServerRegister_KeepAlive_InputMessage

This is the request for the **KeepAlive** operation.

```
<wsdl:message name="IPCBackupServerRegister_KeepAlive_InputMessage">
  <wsdl:part name="parameters" element="tns:KeepAlive" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/KeepAlive
```

The body of the SOAP message MUST contain the **KeepAlive** element.

#### 3.2.5.2.1.2    IPCBackupServerRegister_KeepAlive_OutputMessage

This is the response for the **KeepAlive** operation.

```
<wsdl:message name="IPCBackupServerRegister_KeepAlive_OutputMessage">
  <wsdl:part name="parameters" element="tns:KeepAliveResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/KeepAliveResponse
```

The body of the SOAP message MUST contain the **KeepAliveResponse** element.

### 3.2.5.2.2 Elements

#### 3.2.5.2.2.1    KeepAlive

This element specifies input values for the **KeepAlive** operation.

```
<xs:element name="KeepAlive">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.2.5.2.2.2    KeepAliveResponse

This element specifies output values for the **KeepAlive** operation.

```
<xs:element name="KeepAliveResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.3  IPCBackupServerRegister.VolumeCalculatorProgress

Notifies the server of progress on a volume tree calculation.

```
<wsdl:operation name="VolumeCalculatorProgress">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerRegister/VolumeCalculatorProgress"
message="tns:IPCBackupServerRegister_VolumeCalculatorProgress_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_VolumeCalculatorProgress_InputMessage**
request message, the protocol server is notified of the changes in progress on the volume tree
calculation.

### 3.2.5.3.1 Messages

#### 3.2.5.3.1.1    IPCBackupServerRegister_VolumeCalculatorProgress_InputMessage

This is the request for the **VolumeCalculatorProgress** operation.

```
<wsdl:message name="IPCBackupServerRegister_VolumeCalculatorProgress_InputMessage">
  <wsdl:part name="parameters" element="tns:VolumeCalculatorProgress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/VolumeCalculatorProgress
```

The body of the SOAP message MUST contain the **VolumeCalculatorProgress** element.

### 3.2.5.3.2 Elements

### 3.2.5.3.2.1   VolumeCalculatorProgress

This element specifies input values for the **VolumeCalculatorProgress** operation.

```
<xs:element name="VolumeCalculatorProgress">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="guid"
xmlns:q2="http://schemas.microsoft.com/2003/10/Serialization/" type="q2:guid" />
      <xs:element minOccurs="0" name="percentDone" type="xs:int" />
      <xs:element minOccurs="0" name="folders" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q3:ArrayOfFolderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.4  IPCBackupServerRegister.GetClientPriority

This callback operation returns the priority of the client.

```
<wsdl:operation name="GetClientPriority">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetClientPriority"
message="tns:IPCBackupServerRegister_GetClientPriority_OutputCallbackMessage" />
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetClientPriorityResponse"
message="tns:IPCBackupServerRegister_GetClientPriority_InputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_GetClientPriority_OutputCallbackMessage**
request message, the protocol client MUST return its priority with
**IPCBackupServerRegister_GetClientPriority_InputCallbackMessage**. In the event of a failure,
the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.4.1 Messages

### 3.2.5.4.1.1   IPCBackupServerRegister_GetClientPriority_OutputCallbackMessage

This is the request for the **GetClientPriority** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetClientPriority_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetClientPriority" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/GetClientPriority
```

The body of the SOAP message MUST contain the **GetClientPriority** element.

### 3.2.5.4.1.2   IPCBackupServerRegister_GetClientPriority_InputCallbackMessage

This is the response for the **GetClientPriority** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetClientPriority_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetClientPriorityResponse" />
```

```
       </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IPCBackupServerRegister/GetClientPriorityResponse
```

The body of the SOAP message MUST contain the **GetClientPriorityResponse** element.

### 3.2.5.4.2 Elements

#### 3.2.5.4.2.1   GetClientPriority

This element specifies input values for the **GetClientPriority** operation.

```
    <xs:element name="GetClientPriority">
      <xs:complexType>
        <xs:sequence />
      </xs:complexType>
    </xs:element>
```

#### 3.2.5.4.2.2   GetClientPriorityResponse

This element specifies output values for the **GetClientPriority** operation.

```
    <xs:element name="GetClientPriorityResponse">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" name="GetClientPriorityResult" type="xs:int" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
```

### 3.2.5.5  IPCBackupServerRegister.OnBatteryPower

This callback operation returns a value indicating whether the client is running on battery power.

```
    <wsdl:operation name="OnBatteryPower">
      <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/OnBatteryPower"
    message="tns:IPCBackupServerRegister_OnBatteryPower_OutputCallbackMessage" />
      <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerRegister/OnBatteryPowerResponse"
    message="tns:IPCBackupServerRegister_OnBatteryPower_InputCallbackMessage" />
    </wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_OnBatteryPower_OutputCallbackMessage**
request message, the protocol client checks whether it is running on battery. On successful completion
of the action, the client MUST respond with
**IPCBackupServerRegister_OnBatteryPower_InputCallbackMessage**. In the event of a failure,
the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.5.1 Messages

#### 3.2.5.5.1.1   IPCBackupServerRegister_OnBatteryPower_OutputCallbackMessage

This is the request for the **OnBatteryPower** operation.

```
<wsdl:message name="IPCBackupServerRegister_OnBatteryPower_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:OnBatteryPower" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/OnBatteryPower
```

The body of the SOAP message MUST contain the **OnBatteryPower** element.

### 3.2.5.5.1.2    IPCBackupServerRegister_OnBatteryPower_InputCallbackMessage

This is the response for the **OnBatteryPower** operation.

```
<wsdl:message name="IPCBackupServerRegister_OnBatteryPower_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:OnBatteryPowerResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/OnBatteryPowerResponse
```

The body of the SOAP message MUST contain the **OnBatteryPowerResponse** element.

### 3.2.5.5.2 Elements

### 3.2.5.5.2.1    OnBatteryPower

This element specifies input values for the **OnBatteryPower** operation.

```
<xs:element name="OnBatteryPower">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.5.2.2    OnBatteryPowerResponse

This element specifies output values for the **OnBatteryPower** operation.

```
<xs:element name="OnBatteryPowerResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="OnBatteryPowerResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.6  IPCBackupServerRegister.StartBackup

This callback operation throws an invalid operation fault if the backup job is not in the stopped or started state. Has no effect if the backup job is already started.

```
<wsdl:operation name="StartBackup">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/StartBackup"
message="tns:IPCBackupServerRegister_StartBackup_OutputCallbackMessage" />
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerRegister/StartBackupResponse"
message="tns:IPCBackupServerRegister_StartBackup_InputCallbackMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/StartBackupPCBackupClientFaultFault"
name="PCBackupClientFaultFault"
message="tns:IPCBackupServerRegister_StartBackup_PCBackupClientFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_StartBackup_OutputCallbackMessage** request
message, the protocol client starts a backup. Upon successful completion of a start backup operation,
or if one is already in progress, the client MUST respond with
**IPCBackupServerRegister_StartBackup_InputCallbackMessage**. In the event of a failure, or if
backup is not in a stopped or started state, the appropriate SOAP fault MUST be sent to the server as
specified in section 2.2.2.1.

If backup is automatic, **allowPostpone** specifies whether a queued backup can be postponed if the
automatic backup window is finished. The **allowPostpone** option is ignored if backups are configured
to be started manually.

### 3.2.5.6.1 Messages

### 3.2.5.6.1.1   IPCBackupServerRegister_StartBackup_OutputCallbackMessage

This is the request for the **StartBackup** operation.

```
<wsdl:message name="IPCBackupServerRegister_StartBackup_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:StartBackup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/StartBackup
```

The body of the SOAP message MUST contain the **StartBackup** element.

### 3.2.5.6.1.2   IPCBackupServerRegister_StartBackup_InputCallbackMessage

This is the response for the **StartBackup** operation.

```
<wsdl:message name="IPCBackupServerRegister_StartBackup_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:StartBackupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/StartBackupResponse
```

The body of the SOAP message MUST contain the **StartBackupResponse** element.

### 3.2.5.6.2 Elements

### 3.2.5.6.2.1 StartBackup

This element specifies input values for the **StartBackup** operation.

```xml
<xs:element name="StartBackup">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="automatic" type="xs:boolean" />
      <xs:element minOccurs="0" name="description" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="allowPostpone" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.6.2.2 StartBackupResponse

This element specifies output values for the **StartBackup** operation.

```xml
<xs:element name="StartBackupResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.7 IPCBackupServerRegister.CancelBackup

This callback operation cancels the backup job, regardless of the job's current state.

```xml
<wsdl:operation name="CancelBackup">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/CancelBackup"
message="tns:IPCBackupServerRegister_CancelBackup_OutputCallbackMessage" />
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerRegister/CancelBackupResponse"
message="tns:IPCBackupServerRegister_CancelBackup_InputCallbackMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/CancelBackupPCBackupClientFaultFault"
name="PCBackupClientFaultFault"
message="tns:IPCBackupServerRegister_CancelBackup_PCBackupClientFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_CancelBackup_OutputCallbackMessage** request message, the protocol client cancels the backup job without regard for its state. If it is already idle, then it does nothing. On successful completion of the action, the client MUST respond with **IPCBackupServerRegister_CancelBackup_InputCallbackMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.7.1 Messages

#### 3.2.5.7.1.1 IPCBackupServerRegister_CancelBackup_OutputCallbackMessage

This is the request for the **CancelBackup** operation.

```xml
<wsdl:message name="IPCBackupServerRegister_CancelBackup_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:CancelBackup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/CancelBackup
```

The body of the SOAP message MUST contain the **CancelBackup** element.

### 3.2.5.7.1.2    IPCBackupServerRegister_CancelBackup_InputCallbackMessage

This is the response for the **CancelBackup** operation.

```
<wsdl:message name="IPCBackupServerRegister_CancelBackup_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:CancelBackupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/CancelBackupResponse
```

The body of the SOAP message MUST contain the **CancelBackupResponse** element.

### 3.2.5.7.2 Elements

### 3.2.5.7.2.1    CancelBackup

This element specifies input values for the **CancelBackup** operation.

```
<xs:element name="CancelBackup">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.7.2.2    CancelBackupResponse

This element specifies output values for the **CancelBackup** operation.

```
<xs:element name="CancelBackupResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.8  IPCBackupServerRegister.UpdateBackupConfiguration

This callback operation sets the protocol server to update the backup configuration on the client whenever it changes.

```
<wsdl:operation name="UpdateBackupConfiguration">
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerRegister/UpdateBackupConfiguration"
  message="tns:IPCBackupServerRegister_UpdateBackupConfiguration_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the
**IPCBackupServerRegister_UpdateBackupConfiguration_OutputCallbackMessage** request

message, the protocol client updates the backup configuration on the server whenever it changes. In the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.8.1 Messages

#### 3.2.5.8.1.1    IPCBackupServerRegister_UpdateBackupConfiguration_OutputCallbackMessage

This is the request for the **UpdateBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerRegister_UpdateBackupConfiguration_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:UpdateBackupConfiguration" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/UpdateBackupConfiguration
```

The body of the SOAP message MUST contain the **UpdateBackupConfiguration** element.

### 3.2.5.8.2 Elements

#### 3.2.5.8.2.1    UpdateBackupConfiguration

This element specifies input values for the **UpdateBackupConfiguration** operation.

```
<xs:element name="UpdateBackupConfiguration">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="backupConfig" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q4:ClientConfig" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.9  IPCBackupServerRegister.SendBackupProgress

This callback operation updates the percent complete of a running backup operation.

```
<wsdl:operation name="SendBackupProgress">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/SendBackupProgress"
message="tns:IPCBackupServerRegister_SendBackupProgress_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_SendBackupProgress_OutputCallbackMessage** request message, the protocol client updates the progress percentage of a running backup operation. In the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.9.1 Messages

### 3.2.5.9.1.1 IPCBackupServerRegister_SendBackupProgress_OutputCallbackMessage

This is the request for the **SendBackupProgress** operation.

```
<wsdl:message name="IPCBackupServerRegister_SendBackupProgress_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:SendBackupProgress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/SendBackupProgress
```

The body of the SOAP message MUST contain the **SendBackupProgress** element.

### 3.2.5.9.2 Elements

### 3.2.5.9.2.1 SendBackupProgress

This element specifies input values for the **SendBackupProgress** operation.

```
<xs:element name="SendBackupProgress">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="progress" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.10      IPCBackupServerRegister.GetFolderChildren

This callback operation returns the children of a specific folder.

```
<wsdl:operation name="GetFolderChildren">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetFolderChildren"
message="tns:IPCBackupServerRegister_GetFolderChildren_OutputCallbackMessage" />
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetFolderChildrenResponse"
message="tns:IPCBackupServerRegister_GetFolderChildren_InputCallbackMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetFolderChildrenPCBackupClientFaultF
ault" name="PCBackupClientFaultFault"
message="tns:IPCBackupServerRegister_GetFolderChildren_PCBackupClientFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_GetFolderChildren_OutputCallbackMessage** request message, the protocol client returns the children of the specific folder. It ~~should be~~is called after **CalculateVolume** has finished. On successful completion of the action, the client MUST respond with **IPCBackupServerRegister_GetFolderChildren_InputCallbackMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.10.1      Messages

### 3.2.5.10.1.1 IPCBackupServerRegister_GetFolderChildren_OutputCallbackMessage

This is the request for the **GetFolderChildren** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetFolderChildren_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetFolderChildren" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/GetFolderChildren
```

The body of the SOAP message MUST contain the **GetFolderChildren** element.

### 3.2.5.10.1.2  IPCBackupServerRegister_GetFolderChildren_InputCallbackMessage

This is the response for the **GetFolderChildren** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetFolderChildren_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetFolderChildrenResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/GetFolderChildrenResponse
```

The body of the SOAP message MUST contain the **GetFolderChildrenResponse** element.

### 3.2.5.10.2    Elements

### 3.2.5.10.2.1  GetFolderChildren

This element specifies input values for the **GetFolderChildren** operation.

```
<xs:element name="GetFolderChildren">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="folderRequest" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q5:FolderRequest" />
      <xs:element minOccurs="0" name="index" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.10.2.2  GetFolderChildrenResponse

This element specifies output values for the **GetFolderChildren** operation.

```
<xs:element name="GetFolderChildrenResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetFolderChildrenResult" nillable="true"
xmlns:q6="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q6:FolderInfoGroup" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.11　IPCBackupServerRegister.CalculateVolumeAsync

This callback operation calculates the folder tree for a given volume.

```
<wsdl:operation name="CalculateVolumeAsync">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/CalculateVolumeAsync"
message="tns:IPCBackupServerRegister_CalculateVolumeAsync_OutputCallbackMessage" />
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerRegister/CalculateVolumeAsyncResponse"
message="tns:IPCBackupServerRegister_CalculateVolumeAsync_InputCallbackMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/CalculateVolumeAsyncPCBackupClientFau
ltFault" name="PCBackupClientFaultFault"
message="tns:IPCBackupServerRegister_CalculateVolumeAsync_PCBackupClientFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_CalculateVolumeAsync_OutputCallbackMessage**
request message, the protocol client calculates the folder tree for the volume. On successful
completion of the action, the client MUST respond with
**IPCBackupServerRegister_CalculateVolumeAsync_InputCallbackMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

#### 3.2.5.11.1　Messages

#### 3.2.5.11.1.1　IPCBackupServerRegister_CalculateVolumeAsync_OutputCallbackMessage

This is the request for the **CalculateVolumeAsync** operation.

```
<wsdl:message name="IPCBackupServerRegister_CalculateVolumeAsync_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:CalculateVolumeAsync" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/CalculateVolumeAsync
```

The body of the SOAP message MUST contain the **CalculateVolumeAsync** element.

#### 3.2.5.11.1.2　IPCBackupServerRegister_CalculateVolumeAsync_InputCallbackMessage

This is the response for the **CalculateVolumeAsync** operation.

```
<wsdl:message name="IPCBackupServerRegister_CalculateVolumeAsync_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:CalculateVolumeAsyncResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/CalculateVolumeAsyncResponse
```

The body of the SOAP message MUST contain the **CalculateVolumeAsyncResponse** element.

### 3.2.5.11.2    Elements

### 3.2.5.11.2.1  CalculateVolumeAsync

This element specifies input values for the **CalculateVolumeAsync** operation.

```
<xs:element name="CalculateVolumeAsync">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="volumeGuid"
xmlns:q7="http://schemas.microsoft.com/2003/10/Serialization/" type="q7:guid" />
      <xs:element minOccurs="0" name="excludedFolders" nillable="true"
xmlns:q8="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q8:ArrayOfstring"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.11.2.2  CalculateVolumeAsyncResponse

This element specifies output values for the **CalculateVolumeAsync** operation.

```
<xs:element name="CalculateVolumeAsyncResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="CalculateVolumeAsyncResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.12      IPCBackupServerRegister.CleanFolderSizeCalculation

This callback operation resets the protocol client's folder size calculation.

```
<wsdl:operation name="CleanFolderSizeCalculation">
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerRegister/CleanFolderSizeCalculation"
message="tns:IPCBackupServerRegister_CleanFolderSizeCalculation_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the
**IPCBackupServerRegister_CleanFolderSizeCalculation_OutputCallbackMessage** request
message, the protocol client resets the folder size calculation. In the event of a failure, an appropriate
SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.12.1    Messages

### 3.2.5.12.1.1  IPCBackupServerRegister_CleanFolderSizeCalculation_OutputCallbackM essage

This is the request for the **CleanFolderSizeCalculation** operation.

```
<wsdl:message
name="IPCBackupServerRegister_CleanFolderSizeCalculation_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:CleanFolderSizeCalculation" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/CleanFolderSizeCalculation
```

The body of the SOAP message MUST contain the **CleanFolderSizeCalculation** element.

### 3.2.5.12.2    Elements

### 3.2.5.12.2.1  CleanFolderSizeCalculation

This element specifies input values for the **CleanFolderSizeCalculation** operation.

```
<xs:element name="CleanFolderSizeCalculation">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.13        IPCBackupServerRegister.GetConnectedVolumes

This callback operation is called by the server to obtain the list of volumes connected to the client.

```
<wsdl:operation name="GetConnectedVolumes">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetConnectedVolumes"
message="tns:IPCBackupServerRegister_GetConnectedVolumes_OutputCallbackMessage" />
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetConnectedVolumesResponse"
message="tns:IPCBackupServerRegister_GetConnectedVolumes_InputCallbackMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerRegister/GetConnectedVolumesPCBackupClientFaul
tFault" name="PCBackupClientFaultFault"
message="tns:IPCBackupServerRegister_GetConnectedVolumes_PCBackupClientFaultFault_FaultMessag
e" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_GetConnectedVolumes_OutputCallbackMessage** request message, the protocol client returns the list of configured and unconfigured volumes connected to the client. On successful completion of the action, the client MUST respond with **IPCBackupServerRegister_GetConnectedVolumes_InputCallbackMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.2.5.13.1    Messages

### 3.2.5.13.1.1  IPCBackupServerRegister_GetConnectedVolumes_OutputCallbackMessa ge

This is the request for the **GetConnectedVolumes** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetConnectedVolumes_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetConnectedVolumes" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/GetConnectedVolumes
```

The body of the SOAP message MUST contain the **GetConnectedVolumes** element.

### 3.2.5.13.1.2 IPCBackupServerRegister_GetConnectedVolumes_InputCallbackMessage

This is the response for the **GetConnectedVolumes** operation.

```
<wsdl:message name="IPCBackupServerRegister_GetConnectedVolumes_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:GetConnectedVolumesResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/GetConnectedVolumesResponse
```

The body of the SOAP message MUST contain the **GetConnectedVolumesResponse** element.

### 3.2.5.13.2     Elements

### 3.2.5.13.2.1 GetConnectedVolumes

This element specifies input values for the **GetConnectedVolumes** operation.

```
<xs:element name="GetConnectedVolumes">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.13.2.2 GetConnectedVolumesResponse

This element specifies output values for the **GetConnectedVolumes** operation.

```
<xs:element name="GetConnectedVolumesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetConnectedVolumesResult" nillable="true"
xmlns:q9="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q9:ArrayOfClientVolumeInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.14     IPCBackupClientManagement.RescanVolumesAsync

This callback operation is called by the server to force a client to send the volume configuration to the server.

```
<wsdl:operation name="RescanVolumesAsync">
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerRegister/RescanVolumesAsync"
message="tns:IPCBackupServerRegister_RescanVolumesAsync_OutputCallbackMessage" />
```

```
</wsdl:operation>
```

Upon receiving the **IPCBackupServerRegister_RescanVolumesAsync_OutputCallbackMessage**
request message, the protocol client starts to send the volume configuration to the protocol server. In
the event of a failure, the appropriate SOAP fault MUST be sent to the server as specified in section
2.2.2.1.

### 3.2.5.14.1    Messages

### 3.2.5.14.1.1  IPCBackupServerRegister_RescanVolumesAsync_OutputCallbackMessag e

This is the request for the **RescanVolumesAsync** operation.

```
<wsdl:message name="IPCBackupServerRegister_RescanVolumesAsync_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RescanVolumesAsync" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerRegister/RescanVolumesAsync
```

The body of the SOAP message MUST contain the **RescanVolumesAsync** element.

### 3.2.5.14.2    Elements

### 3.2.5.14.2.1  RescanVolumesAsync

This element specifies input values for the **RescanVolumesAsync** operation.

```
<xs:element name="RescanVolumesAsync">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.15        IPCBackupServerProvider.Register

This operation registers the caller with the PC backup server provider.

```
<wsdl:operation name="Register">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/Register"
message="tns:IPCBackupServerProvider_Register_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/RegisterResponse"
message="tns:IPCBackupServerProvider_Register_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/RegisterSecurityAccessDeniedException
Fault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_Register_SecurityAccessDeniedExceptionFault_FaultMessage
" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_Register_InputMessage** request message, the
protocol server registers the caller with the PC backup server provider. Upon successful completion of

the action, the server MUST respond with **IPCBackupServerProvider_Register_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.15.1    Messages

### 3.2.5.15.1.1  IPCBackupServerProvider_Register_InputMessage

This is the request for the **Register** operation.

```
<wsdl:message name="IPCBackupServerProvider_Register_InputMessage">
  <wsdl:part name="parameters" element="tns:Register" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/Register
```

The body of the SOAP message MUST contain the **Register** element.

### 3.2.5.15.1.2  IPCBackupServerProvider_Register_OutputMessage

This is the response for the **Register** operation.

```
<wsdl:message name="IPCBackupServerProvider_Register_OutputMessage">
  <wsdl:part name="parameters" element="tns:RegisterResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/RegisterResponse
```

The body of the SOAP message MUST contain the **RegisterResponse** element.

### 3.2.5.15.2    Elements

### 3.2.5.15.2.1  Register

This element specifies input values for the **RegisterResponse** operation.

```
<xs:element name="Register">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.15.2.2  RegisterResponse

This element specifies output values for the **RegisterResponse** operation.

```
<xs:element name="RegisterResponse">
  <xs:complexType>
    <xs:sequence />
```

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

### 3.2.5.16        IPCBackupServerProvider.GetClients

This operation returns the server's list of all clients.

```
<wsdl:operation name="GetClients">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClients"
message="tns:IPCBackupServerProvider_GetClients_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientsResponse"
message="tns:IPCBackupServerProvider_GetClients_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientsSecurityAccessDeniedExcepti
onFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetClients_SecurityAccessDeniedExceptionFault_FaultMessa
ge" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientsPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetClients_PCBackupServerFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetClients_InputMessage** request message, the
protocol server generates list of all clients. Upon successful completion of the action, the server MUST
respond with **IPCBackupServerProvider_GetClients_OutputMessage**. In the event of a failure,
the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.16.1      Messages

#### 3.2.5.16.1.1  IPCBackupServerProvider_GetClients_InputMessage

This is the request for the **GetClients** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClients_InputMessage">
  <wsdl:part name="parameters" element="tns:GetClients" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClients
```

The body of the SOAP message MUST contain the **GetClients** element.

#### 3.2.5.16.1.2  IPCBackupServerProvider_GetClients_OutputMessage

This is the response for the **GetClients** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClients_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetClientsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClientsResponse
```

The body of the SOAP message MUST contain the **GetClientsResponse** element.

### 3.2.5.16.2    Elements

#### 3.2.5.16.2.1  GetClients

This element specifies input values for the **GetClients** operation.

```
<xs:element name="GetClients">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.2.5.16.2.2  GetClientsResponse

This element specifies output values for the **GetClients** operation.

```
<xs:element name="GetClientsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetClientsResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q1:ArrayOfClientConfigurationInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.17      IPCBackupServerProvider.GetClient

This operation returns the backup configuration information for a specific client.

```
<wsdl:operation name="GetClient">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClient"
message="tns:IPCBackupServerProvider_GetClient_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientResponse"
message="tns:IPCBackupServerProvider_GetClient_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientSecurityAccessDeniedExceptio
nFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetClient_SecurityAccessDeniedExceptionFault_FaultMessag
e" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetClient_PCBackupServerFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetClient_InputMessage** request message, the
protocol server gets the backup client information of the client. Upon successful completion of the
action, the server MUST respond with **IPCBackupServerProvider_GetClient_OutputMessage**. In
the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section
2.2.2.1.

#### 3.2.5.17.1    Messages

### 3.2.5.17.1.1 IPCBackupServerProvider_GetClient_InputMessage

This is the request for the **GetClient** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClient_InputMessage">
  <wsdl:part name="parameters" element="tns:GetClient" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClient
```

The body of the SOAP message MUST contain the **GetClient** element.

### 3.2.5.17.1.2 IPCBackupServerProvider_GetClient_OutputMessage

This is the response for the **GetClient** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClient_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetClientResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClientResponse
```

The body of the SOAP message MUST contain the **GetClientResponse** element.

### 3.2.5.17.2    Elements

### 3.2.5.17.2.1 GetClient

This element specifies input values for the **GetClient** operation.

```
<xs:element name="GetClient">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.17.2.2 GetClientResponse

This element specifies output values for the **GetClient** operation.

```
<xs:element name="GetClientResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetClientResult" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q2:ClientConfigurationInfo" />
    </xs:sequence>
  </xs:complexType>
```

```
    </xs:element>
```

### 3.2.5.18    IPCBackupServerProvider.GetClientJobs

This operation returns the list of up to 100 client jobs from the server's index.

```
    <wsdl:operation name="GetClientJobs">
      <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientJobs"
message="tns:IPCBackupServerProvider_GetClientJobs_InputMessage" />
      <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientJobsResponse"
message="tns:IPCBackupServerProvider_GetClientJobs_OutputMessage" />
      <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientJobsSecurityAccessDeniedExce
ptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetClientJobs_SecurityAccessDeniedExceptionFault_FaultMe
ssage" />
      <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientJobsPCBackupServerFaultFault
" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetClientJobs_PCBackupServerFaultFault_FaultMessage" />
    </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetClientJobs_InputMessage** request message, the protocol server gets the list of the specified client jobs, or the first 100 if the list contains more than 100 jobs. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_GetClientJobs_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.18.1    Messages

#### 3.2.5.18.1.1 IPCBackupServerProvider_GetClientJobs_InputMessage

This is the request for the **GetClientJobs** operation.

```
    <wsdl:message name="IPCBackupServerProvider_GetClientJobs_InputMessage">
      <wsdl:part name="parameters" element="tns:GetClientJobs" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IPCBackupServerProvider/GetClientJobs
```

The body of the SOAP message MUST contain the **GetClientJobs** element.

#### 3.2.5.18.1.2 IPCBackupServerProvider_GetClientJobs_OutputMessage

This is the response for the **GetClientJobs** operation.

```
    <wsdl:message name="IPCBackupServerProvider_GetClientJobs_OutputMessage">
      <wsdl:part name="parameters" element="tns:GetClientJobsResponse" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClientJobsResponse
```

The body of the SOAP message MUST contain the **GetClientJobsResponse** element.

### 3.2.5.18.2 Elements

#### 3.2.5.18.2.1 GetClientJobs

This element specifies input values for the **GetClientJobs** operation.

```
<xs:element name="GetClientJobs">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="index" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.2.5.18.2.2 GetClientJobsResponse

This element specifies output values for the **GetClientJobs** operation.

```
<xs:element name="GetClientJobsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetClientJobsResult" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q3:ArrayOfClientJobInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.19 IPCBackupServerProvider.GetVolumeJobs

This operation gets the list of all volume jobs.

```
<wsdl:operation name="GetVolumeJobs">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetVolumeJobs"
message="tns:IPCBackupServerProvider_GetVolumeJobs_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetVolumeJobsResponse"
message="tns:IPCBackupServerProvider_GetVolumeJobs_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetVolumeJobsPCBackupServerFaultFault
" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetVolumeJobs_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetVolumeJobsSecurityAccessDeniedExce
ptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetVolumeJobs_SecurityAccessDeniedExceptionFault_FaultMe
ssage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetVolumeJobs_InputMessage** request message, the protocol server generates the list of all volume jobs. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_GetVolumeJobs_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.19.1 Messages

### 3.2.5.19.1.1 IPCBackupServerProvider_GetVolumeJobs_InputMessage

This is the request for the **GetVolumeJobs** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetVolumeJobs_InputMessage">
  <wsdl:part name="parameters" element="tns:GetVolumeJobs" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetVolumeJobs
```

The body of the SOAP message MUST contain the **GetVolumeJobs** element.

### 3.2.5.19.1.2 IPCBackupServerProvider_GetVolumeJobs_OutputMessage

This is the response for the **GetVolumeJobs** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetVolumeJobs_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetVolumeJobsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetVolumeJobsResponse
```

The body of the SOAP message MUST contain the **GetVolumeJobsResponse** element.

### 3.2.5.19.2 Elements

### 3.2.5.19.2.1 GetVolumeJobs

This element specifies input values for the **GetVolumeJobs** operation.

```
<xs:element name="GetVolumeJobs">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="jobIndex" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.19.2.2 GetVolumeJobsResponse

This element specifies output values for the **GetVolumeJobs** operation.

```
<xs:element name="GetVolumeJobsResponse">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="GetVolumeJobsResult" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q4:ArrayOfVolumeJobInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.20    IPCBackupServerProvider.GetClientVolumes

This operation gets the list of all client volumes.

```
<wsdl:operation name="GetClientVolumes">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientVolumes"
message="tns:IPCBackupServerProvider_GetClientVolumes_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientVolumesResponse"
message="tns:IPCBackupServerProvider_GetClientVolumes_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientVolumesSecurityAccessDeniedE
xceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetClientVolumes_SecurityAccessDeniedExceptionFault_Faul
tMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetClientVolumesPCBackupServerFaultFa
ult" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetClientVolumes_PCBackupServerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetClientVolumes_InputMessage** request
message, the protocol server gets the list of all client volumes. Upon successful completion of the
action, the server MUST respond with
**IPCBackupServerProvider_GetClientVolumes_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.20.1    Messages

#### 3.2.5.20.1.1 IPCBackupServerProvider_GetClientVolumes_InputMessage

This is the request for the **GetClientVolumes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClientVolumes_InputMessage">
  <wsdl:part name="parameters" element="tns:GetClientVolumes" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClientVolumes
```

The body of the SOAP message MUST contain the **GetClientVolumes** element.

#### 3.2.5.20.1.2 IPCBackupServerProvider_GetClientVolumes_OutputMessage

This is the response for the **GetClientVolumes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetClientVolumes_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetClientVolumesResponse" />
```

```
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetClientVolumesResponse
```

The body of the SOAP message MUST contain the **GetClientVolumesResponse** element.

### 3.2.5.20.2    Elements

#### 3.2.5.20.2.1  GetClientVolumes

This element specifies input values for the **GetClientVolumes** operation.

```
<xs:element name="GetClientVolumes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.2.5.20.2.2  GetClientVolumesResponse

This element specifies output values for the **GetClientVolumes** operation.

```
<xs:element name="GetClientVolumesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetClientVolumesResult" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q5:ArrayOfClientVolumeInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.21        IPCBackupServerProvider.SetClientJobDescription

This operation sets a description for a client job.

```
<wsdl:operation name="SetClientJobDescription">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobDescription"
message="tns:IPCBackupServerProvider_SetClientJobDescription_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobDescriptionResponse"
message="tns:IPCBackupServerProvider_SetClientJobDescription_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobDescriptionPCBackupServer
FaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_SetClientJobDescription_PCBackupServerFaultFault_FaultMe
ssage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobDescriptionSecurityAccess
DeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_SetClientJobDescription_SecurityAccessDeniedExceptionFau
lt_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_SetClientJobDescription_InputMessage** request message, the protocol server sets the client job description. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_SetClientJobDescription_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.21.1    Messages

#### 3.2.5.21.1.1  IPCBackupServerProvider_SetClientJobDescription_InputMessage

This is the request for the **SetClientJobDescription** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientJobDescription_InputMessage">
  <wsdl:part name="parameters" element="tns:SetClientJobDescription" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientJobDescription
```

The body of the SOAP message MUST contain the **SetClientJobDescription** element.

#### 3.2.5.21.1.2  IPCBackupServerProvider_SetClientJobDescription_OutputMessage

This is the response for the **SetClientJobDescription** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientJobDescription_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetClientJobDescriptionResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientJobDescriptionResponse
```

The body of the SOAP message MUST contain the **SetClientJobDescriptionResponse** element.

### 3.2.5.21.2    Elements

#### 3.2.5.21.2.1  SetClientJobDescription

This element specifies input values for the **SetClientJobDescription** operation.

```
<xs:element name="SetClientJobDescription">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="clientJobIndex" type="xs:int" />
      <xs:element minOccurs="0" name="Description" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.2.5.21.2.2  SetClientJobDescriptionResponse

This element specifies output values for the **SetClientJobDescription** operation.

```
<xs:element name="SetClientJobDescriptionResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.2.5.22    IPCBackupServerProvider.SetClientJobRetentionState

This operation sets the retention state for a client job.

```
<wsdl:operation name="SetClientJobRetentionState">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionState"
message="tns:IPCBackupServerProvider_SetClientJobRetentionState_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionStateResponse"
message="tns:IPCBackupServerProvider_SetClientJobRetentionState_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionStateSecurityAcc
essDeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_SetClientJobRetentionState_SecurityAccessDeniedException
Fault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionStatePCBackupSer
verFaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_SetClientJobRetentionState_PCBackupServerFaultFault_Faul
tMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_SetClientJobRetentionState_InputMessage**
request message, the protocol server gets the client job retention state. Upon successful completion of
the action, the server MUST respond with
**IPCBackupServerProvider_SetClientJobRetentionState_OutputMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.22.1    Messages

#### 3.2.5.22.1.1  IPCBackupServerProvider_SetClientJobRetentionState_InputMessage

This is the request for the **SetClientJobRetentionState** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientJobRetentionState_InputMessage">
  <wsdl:part name="parameters" element="tns:SetClientJobRetentionState" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionState
```

The body of the SOAP message MUST contain the **SetClientJobRetentionState** element.

#### 3.2.5.22.1.2  IPCBackupServerProvider_SetClientJobRetentionState_OutputMessage

This is the response for the **SetClientJobRetentionState** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientJobRetentionState_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetClientJobRetentionStateResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientJobRetentionStateResponse
```

The body of the SOAP message MUST contain the **SetClientJobRetentionStateResponse** element.

### 3.2.5.22.2    Elements

### 3.2.5.22.2.1  SetClientJobRetentionState

This element specifies input values for the **SetClientJobRetentionState** operation.

```
<xs:element name="SetClientJobRetentionState">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="clientJobIndex" type="xs:int" />
      <xs:element minOccurs="0" name="retentionState"
xmlns:q6="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q6:RetentionState" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.22.2.2  SetClientJobRetentionStateResponse

This element specifies output values for the **SetClientJobRetentionState** operation.

```
<xs:element name="SetClientJobRetentionStateResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.23    IPCBackupServerProvider.SetClientVolumeStatus

This operation sets the status of a client volume.

```
<wsdl:operation name="SetClientVolumeStatus">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatus"
message="tns:IPCBackupServerProvider_SetClientVolumeStatus_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatusResponse"
message="tns:IPCBackupServerProvider_SetClientVolumeStatus_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatusPCBackupServerFa
ultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_SetClientVolumeStatus_PCBackupServerFaultFault_FaultMess
age" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatusSecurityAccessDe
niedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_SetClientVolumeStatus_SecurityAccessDeniedExceptionFault
_FaultMessage" />
```

```
    </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_SetClientVolumeStatus_InputMessage** request message, the protocol server sets the client volume status. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_SetClientVolumeStatus_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.23.1    Messages

### 3.2.5.23.1.1 IPCBackupServerProvider_SetClientVolumeStatus_InputMessage

This is the request for the **SetClientVolumeStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientVolumeStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:SetClientVolumeStatus" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatus
```

The body of the SOAP message MUST contain the **SetClientVolumeStatus** element.

### 3.2.5.23.1.2 IPCBackupServerProvider_SetClientVolumeStatus_OutputMessage

This is the response for the **SetClientVolumeStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetClientVolumeStatus_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetClientVolumeStatusResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetClientVolumeStatusResponse
```

The body of the SOAP message MUST contain the **SetClientVolumeStatusResponse** element.

### 3.2.5.23.2    Elements

### 3.2.5.23.2.1 SetClientVolumeStatus

This element specifies input values for the **SetClientVolumeStatus** operation.

```
<xs:element name="SetClientVolumeStatus">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="volumeGuid"
xmlns:q7="http://schemas.microsoft.com/2003/10/Serialization/" type="q7:guid" />
      <xs:element minOccurs="0" name="Status"
xmlns:q8="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProtec
tion.PCBackup.BackupUtil.WcfContracts" type="q8:VolumeStatus" />
      <xs:element minOccurs="0" name="DeleteJobs" type="xs:boolean" />
```

```
        </xs:sequence>
      </xs:complexType>
  </xs:element>
```

### 3.2.5.23.2.2  SetClientVolumeStatusResponse

This element specifies output values for the **SetClientVolumeStatus** operation.

```
<xs:element name="SetClientVolumeStatusResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.2.5.24      IPCBackupServerProvider.SetExcludedFolders

This operation sets excluded folders.

```
<wsdl:operation name="SetExcludedFolders">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetExcludedFolders"
message="tns:IPCBackupServerProvider_SetExcludedFolders_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetExcludedFoldersResponse"
message="tns:IPCBackupServerProvider_SetExcludedFolders_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetExcludedFoldersPCBackupServerFault
Fault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_SetExcludedFolders_PCBackupServerFaultFault_FaultMessage
" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetExcludedFoldersSecurityAccessDenie
dExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_SetExcludedFolders_SecurityAccessDeniedExceptionFault_Fa
ultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_SetExcludedFolders_InputMessage** request
message, the protocol server sets the specified excluded folders. Upon successful completion of the
action, the server MUST respond with
**IPCBackupServerProvider_SetExcludedFolders_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.24.1     Messages

### 3.2.5.24.1.1  IPCBackupServerProvider_SetExcludedFolders_InputMessage

This is the request for the **SetExcludedFolders** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetExcludedFolders_InputMessage">
  <wsdl:part name="parameters" element="tns:SetExcludedFolders" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetExcludedFolders
```

The body of the SOAP message MUST contain the **SetExcludedFolders** element.

### 3.2.5.24.1.2 IPCBackupServerProvider_SetExcludedFolders_OutputMessage

This is the response for the **SetExcludedFolders** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetExcludedFolders_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetExcludedFoldersResponse" />
</wsdl:message>
This message MUST be sent with the following SOAP action.
http://tempuri.org/IPCBackupServerProvider/SetExcludedFoldersResponse
The body of the SOAP message MUST contain the SetExcludedFoldersResponse element.
```

### 3.2.5.24.2    Elements

### 3.2.5.24.2.1 SetExcludedFolders

This element specifies input values for the **SetExcludedFolders** operation.

```
<xs:element name="SetExcludedFolders">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="volumeGuid"
xmlns:q9="http://schemas.microsoft.com/2003/10/Serialization/" type="q9:guid" />
      <xs:element minOccurs="0" name="excludedFolders" nillable="true"
xmlns:q10="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
type="q10:ArrayOfstring" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.24.2.2 SetExcludedFoldersResponse

This element specifies output values for the **SetExcludedFolders** operation.

```
<xs:element name="SetExcludedFoldersResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.25      IPCBackupServerProvider.EnableBackups

This operation enables backups on a client.

```
<wsdl:operation name="EnableBackups">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/EnableBackups"
message="tns:IPCBackupServerProvider_EnableBackups_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/EnableBackupsResponse"
message="tns:IPCBackupServerProvider_EnableBackups_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/EnableBackupsSecurityAccessDeniedExce
ptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_EnableBackups_SecurityAccessDeniedExceptionFault_FaultMe
ssage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/EnableBackupsPCBackupServerFaultFault
```

```
          " name="PCBackupServerFaultFault"
          message="tns:IPCBackupServerProvider_EnableBackups_PCBackupServerFaultFault_FaultMessage" />
        </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_EnableBackups_InputMessage** request message, the protocol server enables backups on a client. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_EnableBackups_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.25.1    Messages

### 3.2.5.25.1.1  IPCBackupServerProvider_EnableBackups_InputMessage

This is the request for the **EnableBackups** operation.

```
    <wsdl:message name="IPCBackupServerProvider_EnableBackups_InputMessage">
      <wsdl:part name="parameters" element="tns:EnableBackups" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IPCBackupServerProvider/EnableBackups
```

The body of the SOAP message MUST contain the **EnableBackups** element.

### 3.2.5.25.1.2  IPCBackupServerProvider_EnableBackups_OutputMessage

This is the response for the **EnableBackups** operation.

```
    <wsdl:message name="IPCBackupServerProvider_EnableBackups_OutputMessage">
      <wsdl:part name="parameters" element="tns:EnableBackupsResponse" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IPCBackupServerProvider/EnableBackupsResponse
```

The body of the SOAP message MUST contain the **EnableBackupsResponse** element.

### 3.2.5.25.2    Elements

### 3.2.5.25.2.1  EnableBackups

This element specifies input values for the **EnableBackups** operation.

```
    <xs:element name="EnableBackups">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
```

### 3.2.5.25.2.2 EnableBackupsResponse

This element specifies output values for the **EnableBackups** operation.

```
<xs:element name="EnableBackupsResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.26    IPCBackupServerProvider.DisableBackups

This operation disables backups on a client.

```
<wsdl:operation name="DisableBackups">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/DisableBackups"
message="tns:IPCBackupServerProvider_DisableBackups_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/DisableBackupsResponse"
message="tns:IPCBackupServerProvider_DisableBackups_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/DisableBackupsSecurityAccessDeniedExc
eptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_DisableBackups_SecurityAccessDeniedExceptionFault_FaultM
essage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/DisableBackupsPCBackupServerFaultFaul
t" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_DisableBackups_PCBackupServerFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_DisableBackups_InputMessage** request message,
the protocol server disables backups on a client. Upon successful completion of the action, the server
MUST respond with **IPCBackupServerProvider_DisableBackups_OutputMessage**. In the event of
a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.26.1    Messages

### 3.2.5.26.1.1 IPCBackupServerProvider_DisableBackups_InputMessage

This is the request for the **DisableBackups** operation.

```
<wsdl:message name="IPCBackupServerProvider_DisableBackups_InputMessage">
  <wsdl:part name="parameters" element="tns:DisableBackups" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/DisableBackups
```

The body of the SOAP message MUST contain the **DisableBackups** element.

### 3.2.5.26.1.2 IPCBackupServerProvider_DisableBackups_OutputMessage

This is the response for the **DisableBackups** operation.

```
<wsdl:message name="IPCBackupServerProvider_DisableBackups_OutputMessage">
  <wsdl:part name="parameters" element="tns:DisableBackupsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/DisableBackupsResponse
```

The body of the SOAP message MUST contain the **DisableBackupsResponse** element.

### 3.2.5.26.2    Elements

### 3.2.5.26.2.1  DisableBackups

This element specifies input values for the **DisableBackups** operation.

```
<xs:element name="DisableBackups">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="deleteAllJobs" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.26.2.2  DisableBackupsResponse

This element specifies output values for the **DisableBackups** operation.

```
<xs:element name="DisableBackupsResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.27        IPCBackupServerProvider.GetBackupConfiguration

This operation returns a system-wide backup configuration, such as backup window and cleanup schedule.

```
<wsdl:operation name="GetBackupConfiguration">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupConfiguration"
message="tns:IPCBackupServerProvider_GetBackupConfiguration_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupConfigurationResponse"
message="tns:IPCBackupServerProvider_GetBackupConfiguration_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupConfigurationPCBackupServerF
aultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetBackupConfiguration_PCBackupServerFaultFault_FaultMes
sage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupConfigurationSecurityAccessD
eniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetBackupConfiguration_SecurityAccessDeniedExceptionFaul
t_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetBackupConfiguration_InputMessage** request message, the protocol server gets the system wide backup configuration. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_GetBackupConfiguration_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.27.1    Messages

#### 3.2.5.27.1.1  IPCBackupServerProvider_GetBackupConfiguration_InputMessage

This is the request for the **GetBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetBackupConfiguration_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBackupConfiguration" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetBackupConfiguration
```

The body of the SOAP message MUST contain the **GetBackupConfiguration** element.

#### 3.2.5.27.1.2  IPCBackupServerProvider_GetBackupConfiguration_OutputMessage

This is the response for the **GetBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetBackupConfiguration_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBackupConfigurationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetBackupConfigurationResponse
```

The body of the SOAP message MUST contain the **GetBackupConfigurationResponse** element.

### 3.2.5.27.2    Elements

#### 3.2.5.27.2.1  GetBackupConfiguration

This element specifies input values for the **GetBackupConfiguration** operation.

```
<xs:element name="GetBackupConfiguration">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.2.5.27.2.2  GetBackupConfigurationResponse

This element specifies output values for the **GetBackupConfiguration** operation.

```
<xs:element name="GetBackupConfigurationResponse">
```

```
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="GetBackupConfigurationResult" nillable="true"
xmlns:q11="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q11:PCBackupConfiguration" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
```

### 3.2.5.28    IPCBackupServerProvider.GetDefaultBackupConfiguration

This operation returns the default backup configuration settings, such as for backup window and cleanup schedule.

```
      <wsdl:operation name="GetDefaultBackupConfiguration">
        <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfiguration"
message="tns:IPCBackupServerProvider_GetDefaultBackupConfiguration_InputMessage" />
        <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfigurationResponse
" message="tns:IPCBackupServerProvider_GetDefaultBackupConfiguration_OutputMessage" />
        <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfigurationSecurity
AccessDeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetDefaultBackupConfiguration_SecurityAccessDeniedExcept
ionFault_FaultMessage" />
        <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfigurationPCBackup
ServerFaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetDefaultBackupConfiguration_PCBackupServerFaultFault_F
aultMessage" />
      </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetDefaultBackupConfiguration_InputMessage** request message, the protocol server gets the default backup configuration. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_GetDefaultBackupConfiguration_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.28.1    Messages

##### 3.2.5.28.1.1 IPCBackupServerProvider_GetDefaultBackupConfiguration_InputMessage

This is the request for the **GetDefaultBackupConfiguration** operation.

```
      <wsdl:message name="IPCBackupServerProvider_GetDefaultBackupConfiguration_InputMessage">
        <wsdl:part name="parameters" element="tns:GetDefaultBackupConfiguration" />
      </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
      http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfiguration
```

The body of the SOAP message MUST contain the **GetDefaultBackupConfiguration** element.

### 3.2.5.28.1.2 IPCBackupServerProvider_GetDefaultBackupConfiguration_OutputMessage

This is the response for the **GetDefaultBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetDefaultBackupConfiguration_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetDefaultBackupConfigurationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetDefaultBackupConfigurationResponse
```

The body of the SOAP message MUST contain the **GetDefaultBackupConfigurationResponse** element.

### 3.2.5.28.2    Elements

### 3.2.5.28.2.1  GetDefaultBackupConfiguration

This element specifies input values for the **GetDefaultBackupConfiguration** operation.

```
<xs:element name="GetDefaultBackupConfiguration">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.28.2.2  GetDefaultBackupConfigurationResponse

This element specifies output values for the **GetDefaultBackupConfiguration** operation.

```
<xs:element name="GetDefaultBackupConfigurationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetDefaultBackupConfigurationResult" nillable="true"
xmlns:q12="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q12:PCBackupConfiguration" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.29      IPCBackupServerProvider.GetBackupOperationStatus

This operation gets the status of a backup operation.

```
<wsdl:operation name="GetBackupOperationStatus">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatus"
message="tns:IPCBackupServerProvider_GetBackupOperationStatus_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatusResponse"
message="tns:IPCBackupServerProvider_GetBackupOperationStatus_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatusSecurityAcces
sDeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
```

```
message="tns:IPCBackupServerProvider_GetBackupOperationStatus_SecurityAccessDeniedExceptionFa
ult_FaultMessage" />
    <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatusPCBackupServe
rFaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetBackupOperationStatus_PCBackupServerFaultFault_FaultM
essage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetBackupOperationStatus_InputMessage**
request message, the protocol server gets the backup operation status. Upon successful completion of
the action, the server MUST respond with
**IPCBackupServerProvider_GetBackupOperationStatus_OutputMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.29.1    Messages

### 3.2.5.29.1.1  IPCBackupServerProvider_GetBackupOperationStatus_InputMessage

This is the request for the **GetBackupOperationStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetBackupOperationStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:GetBackupOperationStatus" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatus
```

The body of the SOAP message MUST contain the **GetBackupOperationStatus** element.

### 3.2.5.29.1.2  IPCBackupServerProvider_GetBackupOperationStatus_OutputMessage

This is the response for the **GetBackupOperationStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetBackupOperationStatus_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetBackupOperationStatusResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetBackupOperationStatusResponse
```

The body of the SOAP message MUST contain the **GetBackupOperationStatusResponse** element.

### 3.2.5.29.2    Elements

### 3.2.5.29.2.1  GetBackupOperationStatus

This element specifies input values for the **GetBackupOperationStatus** operation.

```
<xs:element name="GetBackupOperationStatus">
  <xs:complexType>
    <xs:sequence />
```

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

### 3.2.5.29.2.2 GetBackupOperationStatusResponse

This element specifies output values for the **GetBackupOperationStatus** operation.

```
<xs:element name="GetBackupOperationStatusResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetBackupOperationStatusResult" nillable="true"
xmlns:q13="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q13:OperationStatusResponse" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.30     IPCBackupServerProvider.GetServiceStatus

This operation gets the status of the backup service.

```
<wsdl:operation name="GetServiceStatus">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetServiceStatus"
message="tns:IPCBackupServerProvider_GetServiceStatus_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetServiceStatusResponse"
message="tns:IPCBackupServerProvider_GetServiceStatus_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetServiceStatusSecurityAccessDeniedE
xceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetServiceStatus_SecurityAccessDeniedExceptionFault_Faul
tMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetServiceStatusPCBackupServerFaultFa
ult" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetServiceStatus_PCBackupServerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetServiceStatus_InputMessage** request
message, the protocol server gets the backup service status. Upon successful completion of the
action, the server MUST respond with
**IPCBackupServerProvider_GetServiceStatus_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.30.1     Messages

#### 3.2.5.30.1.1 IPCBackupServerProvider_GetServiceStatus_InputMessage

This is the request for the **GetServiceStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetServiceStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:GetServiceStatus" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetServiceStatus
```

The body of the SOAP message MUST contain the **GetServiceStatus** element.

### 3.2.5.30.1.2 IPCBackupServerProvider_GetServiceStatus_OutputMessage

This is the response for the **GetServiceStatus** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetServiceStatus_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetServiceStatusResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetServiceStatusResponse
```

The body of the SOAP message MUST contain the **GetServiceStatusResponse** element.

### 3.2.5.30.2    Elements

### 3.2.5.30.2.1 GetServiceStatus

This element specifies input values for the **GetServiceStatus** operation.

```
<xs:element name="GetServiceStatus">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.30.2.2 GetServiceStatusResponse

This element specifies output values for the **GetServiceStatus** operation.

```
<xs:element name="GetServiceStatusResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetServiceStatusResult" nillable="true"
xmlns:q14="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q14:ServiceStatusResponse" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.31       IPCBackupServerProvider.GetWaitingOperations

This operation returns a list of pending operations.

```
<wsdl:operation name="GetWaitingOperations">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetWaitingOperations"
message="tns:IPCBackupServerProvider_GetWaitingOperations_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetWaitingOperationsResponse"
message="tns:IPCBackupServerProvider_GetWaitingOperations_OutputMessage" />
```

```
   <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetWaitingOperationsSecurityAccessDen
iedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetWaitingOperations_SecurityAccessDeniedExceptionFault_
FaultMessage" />
   <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetWaitingOperationsPCBackupServerFau
ltFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetWaitingOperations_PCBackupServerFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetWaitingOperations_InputMessage** request
message, the protocol server gets the list of pending operations. Upon successful completion of the
action, the server MUST respond with
**IPCBackupServerProvider_GetWaitingOperations_OutputMessage**. In the event of a failure,
the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.31.1    Messages

### 3.2.5.31.1.1  IPCBackupServerProvider_GetWaitingOperations_InputMessage

This is the request for the **GetWaitingOperations** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetWaitingOperations_InputMessage">
  <wsdl:part name="parameters" element="tns:GetWaitingOperations" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetWaitingOperations
```

The body of the SOAP message MUST contain the **GetWaitingOperations** element.

### 3.2.5.31.1.2  IPCBackupServerProvider_GetWaitingOperations_OutputMessage

This is the response for the **GetWaitingOperations** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetWaitingOperations_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetWaitingOperationsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetWaitingOperationsResponse
```

The body of the SOAP message MUST contain the **GetWaitingOperationsResponse** element.

### 3.2.5.31.2    Elements

### 3.2.5.31.2.1  GetWaitingOperations

This element specifies input values for the **GetWaitingOperations** operation.

```
<xs:element name="GetWaitingOperations">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.31.2.2 GetWaitingOperationsResponse

This element specifies output values for the **GetWaitingOperations** operation.

```
<xs:element name="GetWaitingOperationsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetWaitingOperationsResult" nillable="true"
xmlns:q15="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q15:ArrayOfWaitingOperation" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.32    IPCBackupServerProvider.SetBackupConfiguration

This operation sets system-wide backup configurations.

```
<wsdl:operation name="SetBackupConfiguration">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetBackupConfiguration"
message="tns:IPCBackupServerProvider_SetBackupConfiguration_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetBackupConfigurationResponse"
message="tns:IPCBackupServerProvider_SetBackupConfiguration_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetBackupConfigurationSecurityAccessD
eniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_SetBackupConfiguration_SecurityAccessDeniedExceptionFaul
t_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/SetBackupConfigurationPCBackupServerF
aultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_SetBackupConfiguration_PCBackupServerFaultFault_FaultMes
sage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_SetBackupConfiguration_InputMessage** request message, the protocol server sets the system-wide backup configurations. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_SetBackupConfiguration_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.32.1    Messages

### 3.2.5.32.1.1 IPCBackupServerProvider_SetBackupConfiguration_InputMessage

This is the request for the **SetBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetBackupConfiguration_InputMessage">
  <wsdl:part name="parameters" element="tns:SetBackupConfiguration" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetBackupConfiguration
```

The body of the SOAP message MUST contain the **SetBackupConfiguration** element.

### 3.2.5.32.1.2  IPCBackupServerProvider_SetBackupConfiguration_OutputMessage

This is the response for the **SetBackupConfiguration** operation.

```
<wsdl:message name="IPCBackupServerProvider_SetBackupConfiguration_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetBackupConfigurationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/SetBackupConfigurationResponse
```

The body of the SOAP message MUST contain the **SetBackupConfigurationResponse** element.

### 3.2.5.32.2    Elements

### 3.2.5.32.2.1  SetBackupConfiguration

This element specifies input values for the **SetBackupConfiguration** operation.

```
<xs:element name="SetBackupConfiguration">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="backupConfig" nillable="true"
xmlns:q16="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q16:PCBackupConfiguration" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.32.2.2  SetBackupConfigurationResponse

This element specifies output values for the **SetBackupConfiguration** operation.

```
<xs:element name="SetBackupConfigurationResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.33        IPCBackupServerProvider.GetUsedBytes

This operation returns the total number of bytes used for all backups.

```
<wsdl:operation name="GetUsedBytes">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetUsedBytes"
message="tns:IPCBackupServerProvider_GetUsedBytes_InputMessage" />
```

```
    <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetUsedBytesResponse"
message="tns:IPCBackupServerProvider_GetUsedBytes_OutputMessage" />
    <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetUsedBytesPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetUsedBytes_PCBackupServerFaultFault_FaultMessage" />
    <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetUsedBytesSecurityAccessDeniedExcep
tionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetUsedBytes_SecurityAccessDeniedExceptionFault_FaultMes
sage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetUsedBytes_InputMessage** request message,
the protocol server returns the total number of bytes used for all backups. Upon successful completion
of the action, the server MUST respond with
**IPCBackupServerProvider_GetUsedBytes_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.33.1     Messages

### 3.2.5.33.1.1  IPCBackupServerProvider_GetUsedBytes_InputMessage

This is the request for the **GetUsedBytes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetUsedBytes_InputMessage">
  <wsdl:part name="parameters" element="tns:GetUsedBytes" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetUsedBytes
```

The body of the SOAP message MUST contain the **GetUsedBytes** element.

### 3.2.5.33.1.2  IPCBackupServerProvider_GetUsedBytes_OutputMessage

This is the response for the **GetUsedBytes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetUsedBytes_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetUsedBytesResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetUsedBytesResponse
```

The body of the SOAP message MUST contain the **GetUsedBytesResponse** element.

### 3.2.5.33.2     Elements

### 3.2.5.33.2.1  GetUsedBytes

This element specifies input values for the **GetUsedBytes** operation.

```
<xs:element name="GetUsedBytes">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.33.2.2 GetUsedBytesResponse

This element specifies output values for the **GetUsedBytes** operation.

```
<xs:element name="GetUsedBytesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetUsedBytesResult" type="xs:long" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.34       IPCBackupServerProvider.StartCleanup

This operation starts the cleanup operation.

```
<wsdl:operation name="StartCleanup">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartCleanup"
message="tns:IPCBackupServerProvider_StartCleanup_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartCleanupResponse"
message="tns:IPCBackupServerProvider_StartCleanup_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartCleanupPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StartCleanup_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartCleanupSecurityAccessDeniedExcep
tionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StartCleanup_SecurityAccessDeniedExceptionFault_FaultMes
sage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StartCleanup_InputMessage** request message, the protocol server starts the cleanup operation. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_StartCleanup_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.34.1     Messages

### 3.2.5.34.1.1 IPCBackupServerProvider_StartCleanup_InputMessage

This is the request for the **StartCleanup** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartCleanup_InputMessage">
  <wsdl:part name="parameters" element="tns:StartCleanup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartCleanup
```

The body of the SOAP message MUST contain the **StartCleanup** element.

### 3.2.5.34.1.2 IPCBackupServerProvider_StartCleanup_OutputMessage

This is the response for the **StartCleanup** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartCleanup_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartCleanupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartCleanupResponse
```

The body of the SOAP message MUST contain the StartCleanupResponse element.

### 3.2.5.34.2 Elements

### 3.2.5.34.2.1 StartCleanup

This element specifies input values for the **StartCleanup** operation.

```
<xs:element name="StartCleanup">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.34.2.2 StartCleanupResponse

This element specifies output values for the **StartCleanup** operation.

```
<xs:element name="StartCleanupResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.35 IPCBackupServerProvider.CancelCleanup

This operation cancels the cleanup operation.

```
<wsdl:operation name="CancelCleanup">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelCleanup"
message="tns:IPCBackupServerProvider_CancelCleanup_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelCleanupResponse"
message="tns:IPCBackupServerProvider_CancelCleanup_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelCleanupPCBackupServerFaultFault
" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_CancelCleanup_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelCleanupSecurityAccessDeniedExce
ptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_CancelCleanup_SecurityAccessDeniedExceptionFault_FaultMe
ssage" />
```

```
        </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CancelCleanup_InputMessage** request message, the protocol server cancels the cleanup operation. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_CancelCleanup_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.35.1    Messages

### 3.2.5.35.1.1  IPCBackupServerProvider_CancelCleanup_InputMessage

This is the request for the **CancelCleanup** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelCleanup_InputMessage">
  <wsdl:part name="parameters" element="tns:CancelCleanup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelCleanup
```

The body of the SOAP message MUST contain the **CancelCleanup** element.

### 3.2.5.35.1.2  IPCBackupServerProvider_CancelCleanup_OutputMessage

This is the response for the **CancelCleanup** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelCleanup_OutputMessage">
  <wsdl:part name="parameters" element="tns:CancelCleanupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelCleanupResponse
```

The body of the SOAP message MUST contain the **CancelCleanupResponse** element.

### 3.2.5.35.2    Elements

### 3.2.5.35.2.1  CancelCleanup

This element specifies input values for the **CancelCleanup** operation.

```
<xs:element name="CancelCleanup">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.35.2.2  CancelCleanupResponse

This element specifies output values for the **CancelCleanup** operation.

```
<xs:element name="CancelCleanupResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.36    IPCBackupServerProvider.StartRepair

This operation starts the repair operation.

```
<wsdl:operation name="StartRepair">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartRepair"
message="tns:IPCBackupServerProvider_StartRepair_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartRepairResponse"
message="tns:IPCBackupServerProvider_StartRepair_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartRepairPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StartRepair_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartRepairSecurityAccessDeniedExcept
ionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StartRepair_SecurityAccessDeniedExceptionFault_FaultMess
age" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StartRepair_InputMessage** request message, the protocol server starts the repair operation. Upon successful completion of the action, the server MUST respond with the **IPCBackupServerProvider_StartRepair_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.36.1    Messages

#### 3.2.5.36.1.1  IPCBackupServerProvider_StartRepair_InputMessage

This is the request for the **StartRepair** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartRepair_InputMessage">
  <wsdl:part name="parameters" element="tns:StartRepair" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartRepair
```

The body of the SOAP message MUST contain the **StartRepair** element.

#### 3.2.5.36.1.2  IPCBackupServerProvider_StartRepair_OutputMessage

This is the response for the **StartRepair** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartRepair_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartRepairResponse" />
```

```
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartRepairResponse
```

The body of the SOAP message MUST contain the **StartRepairResponse** element.

### 3.2.5.36.2    Elements

#### 3.2.5.36.2.1  StartRepair

This element specifies input values for the **StartRepair** operation.

```
<xs:element name="StartRepair">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.2.5.36.2.2  StartRepairResponse

This element specifies output values for the **StartRepair** operation.

```
<xs:element name="StartRepairResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.37      IPCBackupServerProvider.CancelRepair

This operation cancels the repair operation.

```
<wsdl:operation name="CancelRepair">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelRepair"
message="tns:IPCBackupServerProvider_CancelRepair_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelRepairResponse"
message="tns:IPCBackupServerProvider_CancelRepair_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelRepairPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_CancelRepair_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelRepairSecurityAccessDeniedExcep
tionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_CancelRepair_SecurityAccessDeniedExceptionFault_FaultMes
sage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CancelRepair_InputMessage** request message, the protocol server cancels the repair operation. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_CancelRepair_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.37.1    Messages

### 3.2.5.37.1.1 IPCBackupServerProvider_CancelRepair_InputMessage

This is the request for the **CancelRepair** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelRepair_InputMessage">
  <wsdl:part name="parameters" element="tns:CancelRepair" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelRepair
```

The body of the SOAP message MUST contain the **CancelRepair** element.

### 3.2.5.37.1.2 IPCBackupServerProvider_CancelRepair_OutputMessage

This is the response for the **CancelRepair** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelRepair_OutputMessage">
  <wsdl:part name="parameters" element="tns:CancelRepairResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelRepairResponse
```

The body of the SOAP message MUST contain the **CancelRepairResponse** element.

### 3.2.5.37.2    Elements

### 3.2.5.37.2.1 CancelRepair

This element specifies input values for the **CancelRepair** operation.

```
<xs:element name="CancelRepair">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.37.2.2 CancelRepairResponse

This element specifies output values for the **CancelRepair** operation.

```
<xs:element name="CancelRepairResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.38     IPCBackupServerProvider.GetLastRepairResult

This operation gets the most recent repair result.

```
<wsdl:operation name="GetLastRepairResult">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetLastRepairResult"
message="tns:IPCBackupServerProvider_GetLastRepairResult_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetLastRepairResultResponse"
message="tns:IPCBackupServerProvider_GetLastRepairResult_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetLastRepairResultSecurityAccessDeni
edExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetLastRepairResult_SecurityAccessDeniedExceptionFault_F
aultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetLastRepairResultPCBackupServerFaul
tFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetLastRepairResult_PCBackupServerFaultFault_FaultMessag
e" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetLastRepairResult_InputMessage** request
message, the protocol server gets the most recent repair result. Upon successful completion of the
action, the server MUST respond with
**IPCBackupServerProvider_GetLastRepairResult_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.38.1     Messages

#### 3.2.5.38.1.1  IPCBackupServerProvider_GetLastRepairResult_InputMessage

This is the request for the **GetLastRepairResult** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetLastRepairResult_InputMessage">
  <wsdl:part name="parameters" element="tns:GetLastRepairResult" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetLastRepairResult
```

The body of the SOAP message MUST contain the **GetLastRepairResult** element.

#### 3.2.5.38.1.2  IPCBackupServerProvider_GetLastRepairResult_OutputMessage

This is the response for the **GetLastRepairResult** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetLastRepairResult_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetLastRepairResultResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetLastRepairResultResponse
```

The body of the SOAP message MUST contain the **GetLastRepairResultResponse** element.

### 3.2.5.38.2     Elements

### 3.2.5.38.2.1 GetLastRepairResult

This element specifies input values for the **GetLastRepairResult** operation.

```
<xs:element name="GetLastRepairResult">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.38.2.2 GetLastRepairResultResponse

This element specifies output values for the **GetLastRepairResult** operation.

```
<xs:element name="GetLastRepairResultResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetLastRepairResultResult" nillable="true"
xmlns:q17="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q17:RepairResult" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.39      IPCBackupServerProvider.StartConsistencyChecker

This operation starts the consistency checking operation.

```
<wsdl:operation name="StartConsistencyChecker">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartConsistencyChecker"
message="tns:IPCBackupServerProvider_StartConsistencyChecker_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartConsistencyCheckerResponse"
message="tns:IPCBackupServerProvider_StartConsistencyChecker_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartConsistencyCheckerPCBackupServer
FaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StartConsistencyChecker_PCBackupServerFaultFault_FaultMe
ssage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartConsistencyCheckerSecurityAccess
DeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StartConsistencyChecker_SecurityAccessDeniedExceptionFau
lt_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StartConsistencyChecker_InputMessage** request message, the protocol server starts the consistency checking operation. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_StartConsistencyChecker_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.39.1    Messages

### 3.2.5.39.1.1 IPCBackupServerProvider_StartConsistencyChecker_InputMessage

This is the request for the **StartConsistencyChecker** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartConsistencyChecker_InputMessage">
  <wsdl:part name="parameters" element="tns:StartConsistencyChecker" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartConsistencyChecker
```

The body of the SOAP message MUST contain the **StartConsistencyChecker** element.

### 3.2.5.39.1.2  IPCBackupServerProvider_StartConsistencyChecker_OutputMessage

This is the response for the **StartConsistencyChecker** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartConsistencyChecker_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartConsistencyCheckerResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartConsistencyCheckerResponse
```

The body of the SOAP message MUST contain the **StartConsistencyCheckerResponse** element.

### 3.2.5.39.2     Elements

### 3.2.5.39.2.1  StartConsistencyChecker

This element specifies input values for the **StartConsistencyChecker** operation.

```
<xs:element name="StartConsistencyChecker">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="fPreempt" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.39.2.2  StartConsistencyCheckerResponse

This element specifies output values for the **StartConsistencyChecker** operation.

```
<xs:element name="StartConsistencyCheckerResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.40      IPCBackupServerProvider.CancelConsistencyChecker

This operation cancels the consistency checking operation.

```
<wsdl:operation name="CancelConsistencyChecker">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelConsistencyChecker"
message="tns:IPCBackupServerProvider_CancelConsistencyChecker_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelConsistencyCheckerResponse"
message="tns:IPCBackupServerProvider_CancelConsistencyChecker_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelConsistencyCheckerSecurityAcces
sDeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_CancelConsistencyChecker_SecurityAccessDeniedExceptionFa
ult_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelConsistencyCheckerPCBackupServe
rFaultFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_CancelConsistencyChecker_PCBackupServerFaultFault_FaultM
essage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CancelConsistencyChecker_InputMessage** request message, the protocol server cancels the consistency checking operation. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_CancelConsistencyChecker_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.40.1    Messages

#### 3.2.5.40.1.1  IPCBackupServerProvider_CancelConsistencyChecker_InputMessage

This is the request for the **CancelConsistencyChecker** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelConsistencyChecker_InputMessage">
  <wsdl:part name="parameters" element="tns:CancelConsistencyChecker" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelConsistencyChecker
```

The body of the SOAP message MUST contain the **CancelConsistencyChecker** element.

#### 3.2.5.40.1.2  IPCBackupServerProvider_CancelConsistencyChecker_OutputMessage

This is the response for the **CancelConsistencyChecker** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelConsistencyChecker_OutputMessage">
  <wsdl:part name="parameters" element="tns:CancelConsistencyCheckerResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelConsistencyCheckerResponse
```

The body of the SOAP message MUST contain the **CancelConsistencyCheckerResponse** element.

### 3.2.5.40.2   Elements

### 3.2.5.40.2.1   CancelConsistencyChecker

This element specifies input values for the **CancelConsistencyChecker** operation.

```
<xs:element name="CancelConsistencyChecker">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.40.2.2   CancelConsistencyCheckerResponse

This element specifies output values for the **CancelConsistencyChecker** operation.

```
<xs:element name="CancelConsistencyCheckerResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.41      IPCBackupServerProvider.Remove

This operation removes a specific client.

```
<wsdl:operation name="Remove">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/Remove"
message="tns:IPCBackupServerProvider_Remove_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/RemoveResponse"
message="tns:IPCBackupServerProvider_Remove_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/RemoveSecurityAccessDeniedExceptionFa
ult" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_Remove_SecurityAccessDeniedExceptionFault_FaultMessage"
/>
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/RemovePCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_Remove_PCBackupServerFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_Remove_InputMessage** request message, the
protocol server removes the specified client. Upon successful completion of the action, the server
MUST respond with **IPCBackupServerProvider_Remove_OutputMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.41.1   Messages

### 3.2.5.41.1.1   IPCBackupServerProvider_Remove_InputMessage

This is the request for the **Remove** operation.

```
<wsdl:message name="IPCBackupServerProvider_Remove_InputMessage">
  <wsdl:part name="parameters" element="tns:Remove" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/Remove
```

The body of the SOAP message MUST contain the **Remove** element.

### 3.2.5.41.1.2 IPCBackupServerProvider_Remove_OutputMessage

This is the response for the **Remove** operation.

```
<wsdl:message name="IPCBackupServerProvider_Remove_OutputMessage">
  <wsdl:part name="parameters" element="tns:RemoveResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/RemoveResponse
```

The body of the SOAP message MUST contain the **RemoveResponse** element.

### 3.2.5.41.2    Elements

### 3.2.5.41.2.1  Remove

This element specifies input values for the **Remove** operation.

```
<xs:element name="Remove">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.41.2.2  RemoveResponse

This element specifies output values for the **Remove** operation.

```
<xs:element name="RemoveResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.42       IPCBackupServerProvider.CalculateVolumeAsync

This operation calculates the specific volume.

```
<wsdl:operation name="CalculateVolumeAsync">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsync"
message="tns:IPCBackupServerProvider_CalculateVolumeAsync_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsyncResponse"
message="tns:IPCBackupServerProvider_CalculateVolumeAsync_OutputMessage" />
```

```
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsyncSecurityAccessDen
iedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_CalculateVolumeAsync_SecurityAccessDeniedExceptionFault_
FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsyncPCBackupServerFau
ltFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_CalculateVolumeAsync_PCBackupServerFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CalculateVolumeAsync_InputMessage** request
message, the protocol server calculates the specific volume. Upon successful completion of the action,
the server MUST respond with
**IPCBackupServerProvider_CalculateVolumeAsync_OutputMessage**. In the event of a failure,
the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.42.1    Messages

### 3.2.5.42.1.1  IPCBackupServerProvider_CalculateVolumeAsync_InputMessage

This is the request for the **CalculateVolumeAsync** operation.

```
<wsdl:message name="IPCBackupServerProvider_CalculateVolumeAsync_InputMessage">
  <wsdl:part name="parameters" element="tns:CalculateVolumeAsync" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsync
```

The body of the SOAP message MUST contain the **CalculateVolumeAsync** element.

### 3.2.5.42.1.2  IPCBackupServerProvider_CalculateVolumeAsync_OutputMessage

This is the response for the **CalculateVolumeAsync** operation.

```
<wsdl:message name="IPCBackupServerProvider_CalculateVolumeAsync_OutputMessage">
  <wsdl:part name="parameters" element="tns:CalculateVolumeAsyncResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CalculateVolumeAsyncResponse
```

The body of the SOAP message MUST contain the **CalculateVolumeAsyncResponse** element.

### 3.2.5.42.2    Elements

### 3.2.5.42.2.1  CalculateVolumeAsync

This element specifies input values for the **CalculateVolumeAsync** operation.

```
<xs:element name="CalculateVolumeAsync">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="volumeGuid"
xmlns:q18="http://schemas.microsoft.com/2003/10/Serialization/" type="q18:guid" />
      <xs:element minOccurs="0" name="excludedFolders" nillable="true"
xmlns:q19="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
type="q19:ArrayOfstring" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.42.2.2  CalculateVolumeAsyncResponse

This element specifies output values for the **CalculateVolumeAsync** operation.

```
<xs:element name="CalculateVolumeAsyncResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="CalculateVolumeAsyncResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.43      IPCBackupServerProvider.GetFolderChildren

This operation gets the children of specified folder.

```
<wsdl:operation name="GetFolderChildren">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetFolderChildren"
message="tns:IPCBackupServerProvider_GetFolderChildren_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetFolderChildrenResponse"
message="tns:IPCBackupServerProvider_GetFolderChildren_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetFolderChildrenSecurityAccessDenied
ExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetFolderChildren_SecurityAccessDeniedExceptionFault_Fau
ltMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetFolderChildrenPCBackupServerFaultF
ault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetFolderChildren_PCBackupServerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetFolderChildren_InputMessage** request
message, the protocol server gets the children of the specified folder. Upon successful completion of
the action, the server MUST respond with
**IPCBackupServerProvider_GetFolderChildren_OutputMessage**. In the event of a failure, the
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.43.1      Messages

### 3.2.5.43.1.1  IPCBackupServerProvider_GetFolderChildren_InputMessage

This is the request for the **GetFolderChildren** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetFolderChildren_InputMessage">
  <wsdl:part name="parameters" element="tns:GetFolderChildren" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetFolderChildren
```

The body of the SOAP message MUST contain the **GetFolderChildren** element.

### 3.2.5.43.1.2  IPCBackupServerProvider_GetFolderChildren_OutputMessage

This is the response for the **GetFolderChildren** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetFolderChildren_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetFolderChildrenResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetFolderChildrenResponse
```

The body of the SOAP message MUST contain the **GetFolderChildrenResponse** element.

### 3.2.5.43.2    Elements

### 3.2.5.43.2.1  GetFolderChildren

This element specifies input values for the **GetFolderChildren** operation.

```
<xs:element name="GetFolderChildren">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="folderRequest" nillable="true"
xmlns:q20="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q20:FolderRequest" />
      <xs:element minOccurs="0" name="index" type="xs:int" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.43.2.2  GetFolderChildrenResponse

This element specifies output values for the **GetFolderChildren** operation.

```
<xs:element name="GetFolderChildrenResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetFolderChildrenResult" nillable="true"
xmlns:q21="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q21:FolderInfoGroup" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.44 IPCBackupServerProvider.GetConnectedVolumes

This operation gets the connected volumes of the specified client.

```
<wsdl:operation name="GetConnectedVolumes">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumes"
message="tns:IPCBackupServerProvider_GetConnectedVolumes_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumesResponse"
message="tns:IPCBackupServerProvider_GetConnectedVolumes_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumesPCBackupServerFaul
tFault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_GetConnectedVolumes_PCBackupServerFaultFault_FaultMessag
e" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumesSecurityAccessDeni
edExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_GetConnectedVolumes_SecurityAccessDeniedExceptionFault_F
aultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_GetConnectedVolumes_InputMessage** request message, the protocol server gets the connected volumes of the client. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_GetConnectedVolumes_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.2.5.44.1 Messages

#### 3.2.5.44.1.1 IPCBackupServerProvider_GetConnectedVolumes_InputMessage

This is the request for the **GetConnectedVolumes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetConnectedVolumes_InputMessage">
  <wsdl:part name="parameters" element="tns:GetConnectedVolumes" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumes
```

The body of the SOAP message MUST contain the **GetConnectedVolumes** element.

#### 3.2.5.44.1.2 IPCBackupServerProvider_GetConnectedVolumes_OutputMessage

This is the response for the **GetConnectedVolumes** operation.

```
<wsdl:message name="IPCBackupServerProvider_GetConnectedVolumes_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetConnectedVolumesResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/GetConnectedVolumesResponse
```

The body of the SOAP message MUST contain the **GetConnectedVolumesResponse** element.

### 3.2.5.44.2   Elements

#### 3.2.5.44.2.1  GetConnectedVolumes

This element specifies input values for the **GetConnectedVolumes** operation.

```
<xs:element name="GetConnectedVolumes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.2.5.44.2.2  GetConnectedVolumesResponse

This element specifies output values for the **GetConnectedVolumes** operation.

```
<xs:element name="GetConnectedVolumesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetConnectedVolumesResult" nillable="true"
xmlns:q22="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q22:ArrayOfClientVolumeInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.45      IPCBackupServerProvider.CleanFolderSizeCalculation

This operation resets the folder size calculation.

```
<wsdl:operation name="CleanFolderSizeCalculation">
  <wsdl:input
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CleanFolderSizeCalculation"
message="tns:IPCBackupServerProvider_CleanFolderSizeCalculation_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CleanFolderSizeCalculation_InputMessage**
request message, the protocol server resets its folder size calculations. Upon successful completion of
the action, the server MUST respond with
**IPCBackupServerProvider_CleanFolderSizeCalculation_OutputMessage**. In the event of a
failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.45.1     Messages

#### 3.2.5.45.1.1  IPCBackupServerProvider_CleanFolderSizeCalculation_InputMessage

This is the request for the **CleanFolderSizeCalculation** operation.

```
<wsdl:message name="IPCBackupServerProvider_CleanFolderSizeCalculation_InputMessage">
  <wsdl:part name="parameters" element="tns:CleanFolderSizeCalculation" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CleanFolderSizeCalculation
```

The body of the SOAP message MUST contain the **CleanFolderSizeCalculation** element.

### 3.2.5.45.2 Elements

#### 3.2.5.45.2.1 CleanFolderSizeCalculation

This element specifies input values for the **CleanFolderSizeCalculation** operation.

```
<xs:element name="CleanFolderSizeCalculation">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.46　　IPCBackupServerProvider.StartBackup

This operation starts a backup operation on a particular client.

```
<wsdl:operation name="StartBackup">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackup"
message="tns:IPCBackupServerProvider_StartBackup_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupResponse"
message="tns:IPCBackupServerProvider_StartBackup_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StartBackup_PCBackupServerFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupSecurityAccessDeniedExcept
ionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StartBackup_SecurityAccessDeniedExceptionFault_FaultMess
age" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StartBackup_InputMessage** request message, the protocol server starts a backup operation on the particular client. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_StartBackup_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.46.1　　Messages

#### 3.2.5.46.1.1 IPCBackupServerProvider_StartBackup_InputMessage

This is the request for the **StartBackup** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartBackup_InputMessage">
  <wsdl:part name="parameters" element="tns:StartBackup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartBackup
```

The body of the SOAP message MUST contain the **StartBackup** element.

### 3.2.5.46.1.2 IPCBackupServerProvider_StartBackup_OutputMessage

This is the response for the **StartBackup** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartBackup_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartBackupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartBackupResponse
```

The body of the SOAP message MUST contain the **StartBackupResponse** element.

### 3.2.5.46.2 Elements

### 3.2.5.46.2.1 StartBackup

This element specifies input values for the **StartBackup** operation.

```
<xs:element name="StartBackup">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientSid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="description" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.46.2.2 StartBackupResponse

This element specifies output values for the **StartBackup** operation.

```
<xs:element name="StartBackupResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.47 IPCBackupServerProvider.CancelBackup

This operation cancels a backup on a particular client.

```
<wsdl:operation name="CancelBackup">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelBackup"
message="tns:IPCBackupServerProvider_CancelBackup_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelBackupResponse"
message="tns:IPCBackupServerProvider_CancelBackup_OutputMessage" />
```

```
     <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelBackupSecurityAccessDeniedExcep
tionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_CancelBackup_SecurityAccessDeniedExceptionFault_FaultMes
sage" />
     <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/CancelBackupPCBackupServerFaultFault"
name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_CancelBackup_PCBackupServerFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_CancelBackup_InputMessage** request message, the protocol server cancels a backup on the particular client. Upon successful completion of the action, the server MUST respond with **IPCBackupServerProvider_CancelBackup_OutputMessage**. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.2.5.47.1    Messages

### 3.2.5.47.1.1  IPCBackupServerProvider_CancelBackup_InputMessage

This is the request for the **CancelBackup** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelBackup_InputMessage">
  <wsdl:part name="parameters" element="tns:CancelBackup" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelBackup
```

The body of the SOAP message MUST contain the **CancelBackup** element.

### 3.2.5.47.1.2  IPCBackupServerProvider_CancelBackup_OutputMessage

This is the response for the **CancelBackup** operation.

```
<wsdl:message name="IPCBackupServerProvider_CancelBackup_OutputMessage">
  <wsdl:part name="parameters" element="tns:CancelBackupResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/CancelBackupResponse
```

The body of the SOAP message MUST contain the **CancelBackupResponse** element.

### 3.2.5.47.2    Elements

### 3.2.5.47.2.1  CancelBackup

This element specifies input values for the **CancelBackup** operation.

```
<xs:element name="CancelBackup">
```

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

### 3.2.5.47.2.2  CancelBackupResponse

This element specifies output values for the **CancelBackup** operation.

```
    <xs:element name="CancelBackupResponse">
      <xs:complexType>
        <xs:sequence />
      </xs:complexType>
    </xs:element>
```

### 3.2.5.48     IPCBackupServerProvider.StartBackupService

This operation starts the backup service.

```
    <wsdl:operation name="StartBackupService">
      <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupService"
message="tns:IPCBackupServerProvider_StartBackupService_InputMessage" />
      <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupServiceResponse"
message="tns:IPCBackupServerProvider_StartBackupService_OutputMessage" />
      <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupServicePCBackupServerFault
Fault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StartBackupService_PCBackupServerFaultFault_FaultMessage
" />
      <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StartBackupServiceSecurityAccessDenie
dExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StartBackupService_SecurityAccessDeniedExceptionFault_Fa
ultMessage" />
    </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StartBackupService_InputMessage** request
message, the protocol server starts the backup service. Upon successful completion of the action, the
server MUST respond with **IPCBackupServerProvider_StartBackupService_OutputMessage**. In
the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section
2.2.2.1.

### 3.2.5.48.1     Messages

### 3.2.5.48.1.1  IPCBackupServerProvider_StartBackupService_InputMessage

This is the request for the **StartBackupService** operation.

```
    <wsdl:message name="IPCBackupServerProvider_StartBackupService_InputMessage">
      <wsdl:part name="parameters" element="tns:StartBackupService" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartBackupService
```

The body of the SOAP message MUST contain the **StartBackupService** element.

### 3.2.5.48.1.2 IPCBackupServerProvider_StartBackupService_OutputMessage

This is the response for the **StartBackupService** operation.

```
<wsdl:message name="IPCBackupServerProvider_StartBackupService_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartBackupServiceResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StartBackupServiceResponse
```

The body of the SOAP message MUST contain the **StartBackupServiceResponse** element.

### 3.2.5.48.2    Elements

### 3.2.5.48.2.1  StartBackupService

This element specifies input values for the **StartBackupService** operation.

```
<xs:element name="StartBackupService">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.48.2.2  StartBackupServiceResponse

This element specifies output values for the **StartBackupService** operation.

```
<xs:element name="StartBackupServiceResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.49      IPCBackupServerProvider.StopBackupService

This operation stops the backup service.

```
<wsdl:operation name="StopBackupService">
  <wsdl:input wsam:Action="http://tempuri.org/IPCBackupServerProvider/StopBackupService"
message="tns:IPCBackupServerProvider_StopBackupService_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StopBackupServiceResponse"
message="tns:IPCBackupServerProvider_StopBackupService_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StopBackupServiceSecurityAccessDenied
ExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IPCBackupServerProvider_StopBackupService_SecurityAccessDeniedExceptionFault_Fau
ltMessage" />
```

```
    <wsdl:fault
wsam:Action="http://tempuri.org/IPCBackupServerProvider/StopBackupServicePCBackupServerFaultF
ault" name="PCBackupServerFaultFault"
message="tns:IPCBackupServerProvider_StopBackupService_PCBackupServerFaultFault_FaultMessage"
/>
  </wsdl:operation>
```

Upon receiving the **IPCBackupServerProvider_StopBackupService_InputMessage** request
message, the protocol server stops the backup service. Upon successful completion of the action, the
server MUST respond with **IPCBackupServerProvider_StopBackupService_OutputMessage**. In
the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section
2.2.2.1.

### 3.2.5.49.1    Messages

### 3.2.5.49.1.1  IPCBackupServerProvider_StopBackupService_InputMessage

This is the request for the **StopBackupService** operation.

```
<wsdl:message name="IPCBackupServerProvider_StopBackupService_InputMessage">
  <wsdl:part name="parameters" element="tns:StopBackupService" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StopBackupService
```

The body of the SOAP message MUST contain the **StopBackupService** element.

### 3.2.5.49.1.2  IPCBackupServerProvider_StopBackupService_OutputMessage

This is the response for the **StopBackupService** operation.

```
<wsdl:message name="IPCBackupServerProvider_StopBackupService_OutputMessage">
  <wsdl:part name="parameters" element="tns:StopBackupServiceResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/StopBackupServiceResponse
```

The body of the SOAP message MUST contain the **StopBackupServiceResponse** element.

### 3.2.5.49.2    Elements

### 3.2.5.49.2.1  StopBackupService

This element specifies input values for the **StopBackupService** operation.

```
<xs:element name="StopBackupService">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="reason"
xmlns:q23="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q23:StopReason" />
      </xs:sequence>
    </xs:complexType>
</xs:element>
```

### 3.2.5.49.2.2  StopBackupServiceResponse

This element specifies output values for the **StopBackupService** operation.

```
<xs:element name="StopBackupServiceResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.2.5.50      IPCBackupServerProvider.NotifyBackupOperationStatusChange

This callback operation notifies the caller of a change in the status of a backup operation.

```
<wsdl:operation name="NotifyBackupOperationStatusChange">
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/NotifyBackupOperationStatusChange"
message="tns:IPCBackupServerProvider_NotifyBackupOperationStatusChange_OutputCallbackMessage"
/>
</wsdl:operation>
```

Upon receiving the
**IPCBackupServerProvider_NotifyBackupOperationStatusChange_OutputCallbackMessage**
request message, the protocol client is notified of a change in the status of a backup operation.

### 3.2.5.50.1      Messages

### 3.2.5.50.1.1  IPCBackupServerProvider_NotifyBackupOperationStatusChange_Output CallbackMessage

This is the request for the **NotifyBackupOperationStatusChange** operation.

```
<wsdl:message
name="IPCBackupServerProvider_NotifyBackupOperationStatusChange_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:NotifyBackupOperationStatusChange" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/NotifyBackupOperationStatusChange
```

The body of the SOAP message MUST contain the NotifyBackupOperationStatusChange element.

### 3.2.5.50.2      Elements

### 3.2.5.50.2.1  NotifyBackupOperationStatusChange

This element specifies input values for the **NotifyBackupOperationStatusChange** operation.

```
<xs:element name="NotifyBackupOperationStatusChange">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="backupOperationStatus" nillable="true"
xmlns:q24="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q24:OperationStatusResponse" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.5.51 IPCBackupServerProvider.VolumeCalculatorProgress

This callback operation notifies the caller of the current progress of a volume calculation.

```
<wsdl:operation name="VolumeCalculatorProgress">
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/VolumeCalculatorProgress"
message="tns:IPCBackupServerProvider_VolumeCalculatorProgress_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the
**IPCBackupServerProvider_VolumeCalculatorProgress_OutputCallbackMessage** request
message, the protocol client is notified of the current progress of a volume calculation.

#### 3.2.5.51.1 Messages

##### 3.2.5.51.1.1 IPCBackupServerProvider_VolumeCalculatorProgress_OutputCallbackMe ssage

This is the request for the **VolumeCalculatorProgress** operation.

```
<wsdl:message name="IPCBackupServerProvider_VolumeCalculatorProgress_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:VolumeCalculatorProgress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/VolumeCalculatorProgress
```

The body of the SOAP message MUST contain the **VolumeCalculatorProgress** element.

#### 3.2.5.51.2 Elements

##### 3.2.5.51.2.1 VolumeCalculatorProgress

This element specifies input values for the **VolumeCalculatorProgress** operation.

```
<xs:element name="VolumeCalculatorProgress">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="guid"
xmlns:q25="http://schemas.microsoft.com/2003/10/Serialization/" type="q25:guid" />
      <xs:element minOccurs="0" name="percentDone" type="xs:int" />
      <xs:element minOccurs="0" name="folders" nillable="true"
xmlns:q26="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q26:ArrayOfFolderInfo" />
```

```
        </xs:sequence>
      </xs:complexType>
    </xs:element>
```

### 3.2.5.52 IPCBackupServerProvider.NotifyServiceStatusChange

This callback operation notifies the caller of a service status change.

```
<wsdl:operation name="NotifyServiceStatusChange">
  <wsdl:output
wsam:Action="http://tempuri.org/IPCBackupServerProvider/NotifyServiceStatusChange"
message="tns:IPCBackupServerProvider_NotifyServiceStatusChange_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the
**IPCBackupServerProvider_NotifyServiceStatusChange_OutputCallbackMessage** request
message, the protocol client is notified of service status change.

#### 3.2.5.52.1 Messages

#### 3.2.5.52.1.1 IPCBackupServerProvider_NotifyServiceStatusChange_OutputCallbackM essage

This is the request for the **NotifyServiceStatusChange** operation.

```
<wsdl:message name="IPCBackupServerProvider_NotifyServiceStatusChange_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:NotifyServiceStatusChange" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IPCBackupServerProvider/NotifyServiceStatusChange
```

The body of the SOAP message MUST contain the **NotifyServiceStatusChange** element.

#### 3.2.5.52.2 Elements

#### 3.2.5.52.2.1 NotifyServiceStatusChange

This element specifies input values for the **NotifyServiceStatusChange** operation.

```
<xs:element name="NotifyServiceStatusChange">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="serviceStatus" nillable="true"
xmlns:q27="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProte
ction.PCBackup.BackupUtil.WcfContracts" type="q27:ServiceStatusResponse" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.2.6 Timer Events

None.

### 3.2.7 Other Local Events

None.

## 3.3 Health Provider Service Contract Details

### 3.3.1 Abstract Data Model

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

#### 3.3.1.1 ADM_ArrayOfKeyValueOfstringArrayOfAlertInfo

This is an array of dictionaries of the alert information for all joined devices.

**Key:** The SID of the device.

**Value:** The alert information of the given alert.

#### 3.3.1.1.1 Data Model

##### 3.3.1.1.1.1 AlertInfo

Note that the AlertInfo data model also has a corresponding SOAP complex type defined in section 2.2.4.3.2.

**ActionPerformed**: The health definition alert action performed when the alert is detected.

**AdditionalInfo**: A store of additional information bound to the alert.

**DateAndTime**: The date and time when the alert was raised.

**Description**: The description of the alert.

**Escalate**: TRUE, if the alert will be escalated to admin through email; otherwise, FALSE.

**FeatureName**: The name of the feature definition to which the health definition belongs.

**FeatureTile**: The title of the health definition for which the alert is raised.

**HealthDefinitionName**: The name of the feature definition for which the alert is raised.

**HealthDefinitionTitle**: The title of the feature definition to which the health definition belongs.

**InstanceId**: The instance ID of the alert.

**IsNew**: TRUE, if the alert being raised is not already present; otherwise, FALSE.

**IsRepairDefined**: TRUE, if the alert is repairable; otherwise, FALSE.

**IsSuppressed**: TRUE, if the alert is suppressed; otherwise, FALSE.

**IsSystem**: TRUE, if an alert is a system alert; otherwise, FALSE.

**LastSuppressedDateTime**: The last time the alert was suppressed.

**MachineId**: Typically the SID of the client computer.

**MachineName**: The name of the machine on which the alert is raised.

**RepairActionPerformed**: The information about the health definition repair action performed when the alert is repaired.

**Severity**: The health status:

- OK: There is no existing alert.

- Information: Informational alert.

- Warning: Warning alert.

- Critical: Critical alert.

- Report: An alert which collects information from the client.

**SupportLink**: The support link of the alert.

**TroubleshootSteps**: The troubleshooting steps of the alert.

**Troubleshooter**: Allow the user to launch the task from the Alert Viewer.

### 3.3.2  Timers

None.

### 3.3.3  Initialization

None.

### 3.3.4  Higher-Layer Triggered Events

None.

### 3.3.5  Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IAlertProviderCallback.RaiseAlertCompleted | Indicates the raise alert has completed. |
| IAlertProviderCallback.ClearAlertCompleted | Indicates the clear alert has completed. |
| IAlertProviderCallback.SuppressAlertCompleted | Indicates the suppress alert has completed. |
| IAlertProviderCallback.RepairAlertCompleted | Indicates the repair alert has completed. |
| IAlertProviderCallback.EvaluateAlertsCompleted | Indicates the health evaluation cycle has completed. |
| IAlertProviderCallback.RaisedAlerts | Provides the alerts raised on a client computer. |
| IAlertProviderCallback.ClearedAlerts | Provides the alerts cleared on a client computer. |
| IAlertSynchCallback.AlertChanged | Indicates a callback for alert changed event. |
| IAlertManagementProviderService.GetAllNetworkAlerts | Gets the list of **AlertInfo** objects representing the alerts currently active in the entire network. |
| IAlertManagementProviderService.ClearAlertOneWay | Clears an alert that is currently active in the network, |

| OperationContract | Description |
|---|---|
|  | either on the primary server or on a client computer. This operation does not wait until the alert is cleared before returning. |
| IAlertManagementProviderService.ClearAlert | Clears an alert that is currently active in the network, either on the primary server or on a client computer. This operation waits until the alert is cleared before returning. |
| IAlertManagementProviderService.SuppressAlertOneWay | Changes the suppress flag on an alert that is currently active in the network, either on the primary server or on a client computer, to the value provided. This operation does not wait until the alert is suppressed before returning. |
| IAlertManagementProviderService.SuppressAlert | Changes the suppress flag on an alert that is currently active in the network, either on the primary server or on a client computer with the value provided. This operation waits until the alert is suppressed before returning. |
| IAlertManagementProviderService.RaiseAlertOneWay | Raises an alert for a health definition defined for a feature. This operation does not wait until the alert is raised before returning. |
| IAlertManagementProviderService.RaiseAlert | Raises an alert for a health definition defined for a feature. This operation waits until the alert is raised before returning. |
| IAlertManagementProviderService.IsAuthorized | Determines whether the connected user has network alert access. This is a provider service running on the primary server. |
| IAlertSynchProviderService.RegisterClientAgent | Registers a client computer for callback with **IAlertSynchProviderService** provider service running on the primary server. |
| IAlertSynchProviderService.SynchronizeAlerts | Synchronizes the list of **AlertInfo** objects representing alerts raised, cleared, updated, or evaluated on a particular machine in the network to the server. |
| IAlertSynchProviderService.ClearCallbackCommandQueue | Clears all the pending command queues for the machine. Used when a device cannot have an Agent running. |
| IAlertSynchProviderService.GetPlugInInformation | Get information about all the installed plugins. |
| IAlertSynchProviderService.DownloadDefinition | Get stream of definition file of a feature plug-in. |
| IAlertSynchProviderService.DownloadConfiguration | Get stream of definition configuration file of a feature plug-in. |

### 3.3.5.1  IAlertProviderCallback.RaiseAlertCompleted

This operation is a callback indicating that the raise alert operation has completed.

```
<wsdl:operation name="RaiseAlertCompleted">
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlertCompleted"
message="tns:IAlertManagementProviderService_RaiseAlertCompleted_OutputCallbackMessage" />
```

```
    </wsdl:operation>
```

### 3.3.5.1.1 Messages

#### 3.3.5.1.1.1    IAlertProviderCallback_RaiseAlertCompleted_OutputCallbackMessage

This is the request for the **RaiseAlertCompleted** operation.

```
<wsdl:message
name="IAlertManagementProviderService_RaiseAlertCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RaiseAlertCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RaiseAlertCompleted
```

The body of the SOAP message MUST contain the **RaiseAlertCompleted** element.

### 3.3.5.1.2 Elements

#### 3.3.5.1.2.1    RaiseAlertCompleted

This element specifies input values for the **RaiseAlertCompleted** operation.

```
<xs:element name="RaiseAlertCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertInfo" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q5:AlertInfo" />
      <xs:element minOccurs="0" name="errMessage" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

For details on **AlertInfo**, see section 2.2.4.3.2.

### 3.3.5.2  IAlertProviderCallback.ClearAlertCompleted

This operation is a callback indicating that the clear alert operation has completed.

```
<wsdl:operation name="ClearAlertCompleted">
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlertCompleted"
message="tns:IAlertManagementProviderService_ClearAlertCompleted_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.2.1 Messages

#### 3.3.5.2.1.1    IAlertProviderCallback_ClearAlertCompleted_OutputCallbackMessage

This is the request for the **ClearAlertCompleted** operation.

```
<wsdl:message name="IAlertProviderCallback_ClearAlertCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ClearAlertCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/ClearAlertCompleted
```

The body of the SOAP message MUST contain the **ClearAlertCompleted** element.

### 3.3.5.2.2 Elements

#### 3.3.5.2.2.1    ClearAlertCompleted

This element specifies input values for the **ClearAlertCompleted** operation.

```
<xs:element name="ClearAlertCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="errMessage" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.3  IAlertProviderCallback.SuppressAlertCompleted

This operation is a callback indicating that the suppress alert operation has completed.

```
<wsdl:operation name="SuppressAlertCompleted">
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlertCompleted"
message="tns:IAlertManagementProviderService_SuppressAlertCompleted_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.3.1 Messages

#### 3.3.5.3.1.1    IAlertProviderCallback_SuppressAlertCompleted_OutputCallbackMessage

This is the request for the **SuppressAlertCompleted** operation.

```
<wsdl:message name="IAlertProviderCallback_SuppressAlertCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:SuppressAlertCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/SuppressAlertCompleted
```

The body of the SOAP message MUST contain the **SuppressAlertCompleted** element.

### 3.3.5.3.2 Elements

### 3.3.5.3.2.1 SuppressAlertCompleted

This element specifies input values for the **SuppressAlertCompleted** operation.

```
<xs:element name="SuppressAlertCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="errMessage" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.3.5.4 IAlertProviderCallback.RepairAlertCompleted

This operation is a callback indicating that the repair alert operation has completed.

```
<wsdl:operation name="RepairAlertCompleted">
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RepairAlertCompleted"
message="tns:IAlertManagementProviderService_RepairAlertCompleted_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.4.1 Messages

### 3.3.5.4.1.1 IAlertProviderCallback_RepairAlertCompleted_OutputCallbackMessage

This is the request for the **RepairAlertCompleted** operation.

```
<wsdl:message name="IAlertProviderCallback_RepairAlertCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RepairAlertCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RepairAlertCompleted
```

The body of the SOAP message MUST contain the **RepairAlertCompleted** element.

### 3.3.5.4.2 Elements

### 3.3.5.4.2.1 RepairAlertCompleted

This element specifies input values for the **RepairAlertCompleted** operation.

```
<xs:element name="RepairAlertCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="errMessage" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.5 IAlertProviderCallback.EvaluateAlertsCompleted

This operation is a callback indicating that the health evaluation process has completed.

```
<wsdl:operation name="EvaluateAlertsCompleted">
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/EvaluateAlertsCompleted"
message="tns:IAlertManagementProviderService_EvaluateAlertsCompleted_OutputCallbackMessage"
/>
</wsdl:operation>
```

### 3.3.5.5.1 Messages

### 3.3.5.5.1.1    IAlertProviderCallback_EvaluateAlertsCompleted_OutputCallbackMessage

This is the request for the **EvaluateAlertsCompleted** operation.

```
<wsdl:message name="IAlertProviderCallback_EvaluateAlertsCompleted_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:EvaluateAlertsCompleted" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/EvaluateAlertsCompleted
```

The body of the SOAP message MUST contain the **EvaluateAlertsCompleted** element.

### 3.3.5.5.2 Elements

### 3.3.5.5.2.1    EvaluateAlertsCompleted

This element specifies input values for the **EvaluateAlertsCompleted** operation.

```
<xs:element name="EvaluateAlertsCompleted">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="machineId" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="ignoreSchedule" nillable="true" type="xs:boolean" />
      <xs:element minOccurs="0" name="errMessage" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.6 IAlertProviderCallback.RaisedAlerts

This operation is a callback that provides the alerts that have been raised on a client computer.

```
<wsdl:operation name="RaisedAlerts">
  <wsdl:output wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaisedAlerts"
message="tns:IAlertManagementProviderService_RaisedAlerts_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.6.1 Messages

### 3.3.5.6.1.1    IAlertProviderCallback_RaisedAlerts_OutputCallbackMessage

This is the request for the **RaisedAlerts** operation.

```
<wsdl:message name="IAlertProviderCallback_RaisedAlerts_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RaisedAlerts" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RaisedAlerts
```

The body of the SOAP message MUST contain the **RaisedAlerts** element.

### 3.3.5.6.2 Elements

### 3.3.5.6.2.1    RaisedAlerts

This element specifies input values for the **RaisedAlerts** operation.

```
<xs:element name="RaisedAlerts">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertInfoList" nillable="true"
xmlns:q6="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q6:ArrayOfAlertInfo" />
      <xs:element minOccurs="0" name="task"
xmlns:q7="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q7:TaskType" />
      <xs:element minOccurs="0" name="machineId" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.7  IAlertProviderCallback.ClearedAlerts

This operation is a callback that provides the alerts that have been cleared on a client computer.

```
<wsdl:operation name="ClearedAlerts">
  <wsdl:output wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearedAlerts"
message="tns:IAlertManagementProviderService_ClearedAlerts_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.7.1 Messages

### 3.3.5.7.1.1    IAlertProviderCallback_ClearedAlerts_OutputCallbackMessage

This is the request for the **ClearedAlerts** operation.

```
<wsdl:message name="IAlertProviderCallback_ClearedAlerts_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ClearedAlerts" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/ClearedAlerts
```

The body of the SOAP message MUST contain the **ClearedAlerts** element.

### 3.3.5.7.2 Elements

#### 3.3.5.7.2.1    ClearedAlerts

This element specifies input values for the **ClearedAlerts** operation.

```
<xs:element name="ClearedAlerts">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertInfoList" nillable="true"
xmlns:q8="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q8:ArrayOfAlertInfo" />
      <xs:element minOccurs="0" name="task"
xmlns:q9="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q9:TaskType" />
      <xs:element minOccurs="0" name="machineId" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.8  IAlertSynchCallback.AlertChanged

This operation is a callback for the alert changed event.

```
<wsdl:operation name="AlertChanged">
  <wsdl:output wsam:Action="http://tempuri.org/IAlertSynchProviderService/AlertChanged"
message="tns:IAlertSynchProviderService_AlertChanged_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.3.5.8.1 Messages

#### 3.3.5.8.1.1    IAlertSynchCallback_AlertChanged_OutputCallbackMessage

This is the request for the **AlertChanged** operation.

```
<wsdl:message name="IAlertSynchCallback_AlertChanged_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:AlertChanged" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderCallback/AlertChanged
```

The body of the SOAP message MUST contain the **AlertChanged** element.

### 3.3.5.8.2 Elements

#### 3.3.5.8.2.1    AlertChanged

This element specifies input values for the **AlertChanged** operation.

```
<xs:element name="AlertChanged">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="commands" nillable="true"
xmlns:q6="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q6:ArrayOfanyType"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.9  IAlertManagementProviderService.GetAllNetworkAlerts

This operation returns the list of **AlertInfo** objects representing the alerts currently active in the entire network.

```
<wsdl:operation name="GetAllNetworkAlerts">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlerts"
message="tns:IAlertManagementProviderService_GetAllNetworkAlerts_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlertsResponse"
message="tns:IAlertManagementProviderService_GetAllNetworkAlerts_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlertsSecurityAc
cessDeniedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IAlertManagementProviderService_GetAllNetworkAlerts_SecurityAccessDeniedExceptio
nFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlertsStringFaul
t" name="StringFault"
message="tns:IAlertManagementProviderService_GetAllNetworkAlerts_StringFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_GetAllNetworkAlerts_InputMessage** request message, the protocol server starts to query the alerts currently active in the entire network. Upon successful completion of the action, the server MUST respond with **IAlertManagementProviderService_GetAllNetworkAlerts_OutputMessage**, which includes the list of AlertInfo objects representing the results. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.3.5.9.1 Messages

##### 3.3.5.9.1.1    IAlertManagementProviderService_GetAllNetworkAlerts_InputMessage

This is the request for the **GetAllNetworkAlerts** operation.

```
<wsdl:message name="IAlertManagementProviderService_GetAllNetworkAlerts_InputMessage">
  <wsdl:part name="parameters" element="tns:GetAllNetworkAlerts" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlerts
```

The body of the SOAP message MUST contain the **GetAllNetworkAlerts** element.

##### 3.3.5.9.1.2    IAlertManagementProviderService_GetAllNetworkAlerts_OutputMessage

This is the response for the **GetAllNetworkAlerts** operation.

```
<wsdl:message name="IAlertManagementProviderService_GetAllNetworkAlerts_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetAllNetworkAlertsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/GetAllNetworkAlertsResponse
```

The body of the SOAP message MUST contain the **GetAllNetworkAlertsResponse** element.

### 3.3.5.9.2 Elements

#### 3.3.5.9.2.1   GetAllNetworkAlerts

This element specifies input values for the **GetAllNetworkAlerts** operation.

```
<xs:element name="GetAllNetworkAlerts">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.3.5.9.2.2   GetAllNetworkAlertsResponse

This element specifies output values for the **GetAllNetworkAlerts** operation.

```
<xs:element name="GetAllNetworkAlertsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetAllNetworkAlertsResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework.Service" type="q1:AlertListChunk" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.10      IAlertManagementProviderService.ClearAlertOneWay

Clears an alert that is currently active in the network, either on the primary server or on a client computer. This operation does not wait for the alert to be cleared before returning.

```
<wsdl:operation name="ClearAlertOneWay">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlertOneWay"
message="tns:IAlertManagementProviderService_ClearAlertOneWay_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_ClearAlertOneWay_InputMessage**
request message, the protocol server is notified to clear the specific alert.

#### 3.3.5.10.1     Messages

### 3.3.5.10.1.1 IAlertManagementProviderService_ClearAlertOneWay_InputMessage

This is the request for the **ClearAlertOneWay** operation.

```
<wsdl:message name="IAlertManagementProviderService_ClearAlertOneWay_InputMessage">
  <wsdl:part name="parameters" element="tns:ClearAlertOneWay" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/ClearAlertOneWay
```

The body of the SOAP message MUST contain the **ClearAlertOneWay** element.

### 3.3.5.10.2    Elements

### 3.3.5.10.2.1 ClearAlertOneWay

This element specifies input values for the **ClearAlertOneWay** operation.

```
<xs:element name="ClearAlertOneWay">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.11    IAlertManagementProviderService.ClearAlert

This operation clears an alert that is currently active in the network, either on the primary server or on a client computer. This operation waits until the alert is cleared before returning.

```
<wsdl:operation name="ClearAlert">
  <wsdl:input wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlert"
message="tns:IAlertManagementProviderService_ClearAlert_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlertResponse"
message="tns:IAlertManagementProviderService_ClearAlert_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlertStringFault"
name="StringFault"
message="tns:IAlertManagementProviderService_ClearAlert_StringFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/ClearAlertSecurityAccessDenie
dExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IAlertManagementProviderService_ClearAlert_SecurityAccessDeniedExceptionFault_Fa
ultMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_ClearAlert_InputMessage** request message, the protocol server starts to clear the specific alert. Upon successful completion of the action, the server MUST respond with **IAlertManagementProviderService_ClearAlert_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.11.1    Messages

### 3.3.5.11.1.1  IAlertManagementProviderService_ClearAlert_InputMessage

This is the request for the **ClearAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_ClearAlert_InputMessage">
  <wsdl:part name="parameters" element="tns:ClearAlert" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/ClearAlert
```

The body of the SOAP message MUST contain the **ClearAlert** element.

### 3.3.5.11.1.2  IAlertManagementProviderService_ClearAlert_OutputMessage

This is the response for the **ClearAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_ClearAlert_OutputMessage">
  <wsdl:part name="parameters" element="tns:ClearAlertResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/ClearAlertResponse
```

The body of the SOAP message MUST contain the **ClearAlertResponse** element.

### 3.3.5.11.2    Elements

### 3.3.5.11.2.1  ClearAlert

This element specifies input values for the **ClearAlert** operation.

```
<xs:element name="ClearAlert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.11.2.2  ClearAlertResponse

This element specifies output values for the **ClearAlert** operation.

```
<xs:element name="ClearAlertResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.3.5.12    IAlertManagementProviderService.SuppressAlertOneWay

This operation sets the suppress flag on an alert that is currently active in the network to the value provided. The alert can be either on the primary server or on a client computer. This operation does not wait until the alert is suppressed before returning.

```
<wsdl:operation name="SuppressAlertOneWay">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlertOneWay"
message="tns:IAlertManagementProviderService_SuppressAlertOneWay_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_SuppressAlertOneWay_InputMessage** request message, the protocol server is notified to suppress the specific alert.

#### 3.3.5.12.1    Messages

#### 3.3.5.12.1.1  IAlertManagementProviderService_SuppressAlertOneWay_InputMessage

This is the request for the **SuppressAlertOneWay** operation.

```
<wsdl:message name="IAlertManagementProviderService_SuppressAlertOneWay_InputMessage">
  <wsdl:part name="parameters" element="tns:SuppressAlertOneWay" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/SuppressAlertOneWay
```

The body of the SOAP message MUST contain the **SuppressAlertOneWay** element.

#### 3.3.5.12.2    Elements

#### 3.3.5.12.2.1  SuppressAlertOneWay

This element specifies input values for the **SuppressAlertOneWay** operation.

```
<xs:element name="SuppressAlertOneWay">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="value" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.13    IAlertManagementProviderService.SuppressAlert

This operation sets the suppress flag on an alert that is currently active in the network to the value provided. The alert can be either on the primary server or on a client computer. This operation waits until the alert is cleared before returning.

```
<wsdl:operation name="SuppressAlert">
```

```
    <wsdl:input wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlert"
message="tns:IAlertManagementProviderService_SuppressAlert_InputMessage" />
    <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlertResponse"
message="tns:IAlertManagementProviderService_SuppressAlert_OutputMessage" />
    <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlertStringFault"
name="StringFault"
message="tns:IAlertManagementProviderService_SuppressAlert_StringFault_FaultMessage" />
    <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/SuppressAlertSecurityAccessDe
niedExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IAlertManagementProviderService_SuppressAlert_SecurityAccessDeniedExceptionFault
_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_SuppressAlert_InputMessage** request message, the protocol server starts to suppress the specific alert. Upon successful completion of the action, the server MUST respond with **IAlertManagementProviderService_SuppressAlert_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.13.1  Messages

#### 3.3.5.13.1.1  IAlertManagementProviderService_SuppressAlert_InputMessage

This is the request for the **SuppressAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_SuppressAlert_InputMessage">
  <wsdl:part name="parameters" element="tns:SuppressAlert" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/SuppressAlert
```

The body of the SOAP message MUST contain the **SuppressAlert** element.

#### 3.3.5.13.1.2  IAlertManagementProviderService_SuppressAlert_OutputMessage

This is the response for the **SuppressAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_SuppressAlert_OutputMessage">
  <wsdl:part name="parameters" element="tns:SuppressAlertResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/SuppressAlertResponse
```

The body of the SOAP message MUST contain the **SuppressAlertResponse** element.

### 3.3.5.13.2  Elements

#### 3.3.5.13.2.1  SuppressAlert

This element specifies input values for the **SuppressAlert** operation.

```
<xs:element name="SuppressAlert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertKey" nillable="true"
type="xs:string" />
      <xs:element minOccurs="0" name="value" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.13.2.2  SuppressAlertResponse

This element specifies input values for the **SuppressAlert** operation.

```
<xs:element name="SuppressAlertResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.3.5.14      IAlertManagementProviderService.RaiseAlertOneWay

This operation raises an alert for a health definition for a feature. This operation does not wait until the alert is raised before returning.

```
<wsdl:operation name="RaiseAlertOneWay">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlertOneWay"
message="tns:IAlertManagementProviderService_RaiseAlertOneWay_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_RaiseAlertOneWay_InputMessage** request message, the protocol server is notified to raise the specific alert.

### 3.3.5.14.1     Messages

### 3.3.5.14.1.1  IAlertManagementProviderService_RaiseAlertOneWay_InputMessage

This is the request for the **RaiseAlertOneWay** operation.

```
<wsdl:message name="IAlertManagementProviderService_RaiseAlertOneWay_InputMessage">
  <wsdl:part name="parameters" element="tns:RaiseAlertOneWay" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RaiseAlertOneWay
```

The body of the SOAP message MUST contain the **RaiseAlertOneWay** element.

### 3.3.5.14.2     Elements

### 3.3.5.14.2.1 RaiseAlertOneWay

This element specifies input values for the **RaiseAlertOneWay** operation.

```
<xs:element name="RaiseAlertOneWay">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertInfo" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q2:AlertInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.15        IAlertManagementProviderService.RaiseAlert

This operation raises an alert for a health definition for a feature. This operation waits until the alert is cleared before returning.

```
<wsdl:operation name="RaiseAlert">
  <wsdl:input wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlert"
message="tns:IAlertManagementProviderService_RaiseAlert_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlertResponse"
message="tns:IAlertManagementProviderService_RaiseAlert_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlertStringFault"
name="StringFault"
message="tns:IAlertManagementProviderService_RaiseAlert_StringFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertManagementProviderService/RaiseAlertSecurityAccessDenie
dExceptionFault" name="SecurityAccessDeniedExceptionFault"
message="tns:IAlertManagementProviderService_RaiseAlert_SecurityAccessDeniedExceptionFault_Fa
ultMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_RaiseAlert_InputMessage** request message, the protocol server starts to raise the specific alert. Upon successful completion of the action, the server MUST respond with **IAlertManagementProviderService_RaiseAlert_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.3.5.15.1    Messages

##### 3.3.5.15.1.1 IAlertManagementProviderService_RaiseAlert_InputMessage

This is the request for the **RaiseAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_RaiseAlert_InputMessage">
  <wsdl:part name="parameters" element="tns:RaiseAlert" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RaiseAlert
```

The body of the SOAP message MUST contain the **RaiseAlert** element.

### 3.3.5.15.1.2 IAlertManagementProviderService_RaiseAlert_OutputMessage

This is the response for the **RaiseAlert** operation.

```
<wsdl:message name="IAlertManagementProviderService_RaiseAlert_OutputMessage">
  <wsdl:part name="parameters" element="tns:RaiseAlertResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/RaiseAlertResponse
```

The body of the SOAP message MUST contain the **RaiseAlertResponse** element.

### 3.3.5.15.2    Elements

### 3.3.5.15.2.1  RaiseAlert

This element specifies input values for the **RaiseAlert** operation.

```
<xs:element name="RaiseAlert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alertInfo" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q3:AlertInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

For details on **AlertInfo**, see section 2.2.4.3.2.

### 3.3.5.15.2.2  RaiseAlertResponse

This element specifies output values for the **RaiseAlert** operation.

```
<xs:element name="RaiseAlertResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="RaiseAlertResult" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q4:AlertInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.16        IAlertManagementProviderService.IsAuthorized

This operation determines whether the connected user has network alert access.

```
<wsdl:operation name="IsAuthorized">
  <wsdl:input wsam:Action="http://tempuri.org/IAlertManagementProviderService/IsAuthorized"
message="tns:IAlertManagementProviderService_IsAuthorized_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertManagementProviderService/IsAuthorizedResponse"
message="tns:IAlertManagementProviderService_IsAuthorized_OutputMessage" />
```

```
</wsdl:operation>
```

Upon receiving the **IAlertManagementProviderService_IsAuthorized_InputMessage** request message, the protocol server starts to check whether the connected user has network alert access. Upon successful completion of the action, the server MUST respond with **IAlertManagementProviderService_IsAuthorized_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.16.1    Messages

### 3.3.5.16.1.1  IAlertManagementProviderService_IsAuthorized_InputMessage

This is the request for the **IsAuthorized** operation.

```
<wsdl:message name="IAlertManagementProviderService_IsAuthorized_InputMessage">
  <wsdl:part name="parameters" element="tns:IsAuthorized" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/IsAuthorized
```

The body of the SOAP message MUST contain the **IsAuthorized** element.

### 3.3.5.16.1.2  IAlertManagementProviderService_IsAuthorized_OutputMessage

This is the response for the **IsAuthorized** operation.

```
<wsdl:message name="IAlertManagementProviderService_IsAuthorized_OutputMessage">
  <wsdl:part name="parameters" element="tns:IsAuthorizedResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertManagementProviderService/IsAuthorizedResponse
```

The body of the SOAP message MUST contain the **IsAuthorizedResponse** element.

### 3.3.5.16.2    Elements

### 3.3.5.16.2.1  IsAuthorized

This element specifies input values for the **IsAuthorized** operation.

```
<xs:element name="IsAuthorized">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.3.5.16.2.2  IsAuthorizedResponse

This element specifies output values for the **IsAuthorized** operation.

```
<xs:element name="IsAuthorizedResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" name="returnValue" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
```

### 3.3.5.17      IAlertSynchProviderService.RegisterClientAgent

This operation registers a client computer for callbacks with the **IAlertSynchProviderService** instance running on the primary server.

```
<wsdl:operation name="RegisterClientAgent">
  <wsdl:input wsam:Action="http://tempuri.org/IAlertSynchProviderService/RegisterClientAgent"
message="tns:IAlertSynchProviderService_RegisterClientAgent_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertSynchProviderService/RegisterClientAgentResponse"
message="tns:IAlertSynchProviderService_RegisterClientAgent_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAlertSynchProviderService/RegisterClientAgentArgumentNullExc
eptionFault" name="ArgumentNullExceptionFault"
message="tns:IAlertSynchProviderService_RegisterClientAgent_ArgumentNullExceptionFault_FaultM
essage" />
</wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_RegisterClientAgent_InputMessage** request message, the protocol server starts to register the client computer. Upon successful completion of the action, the server MUST respond with **IAlertSynchProviderService_RegisterClientAgent_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.3.5.17.1    Messages

#### 3.3.5.17.1.1 IAlertSynchProviderService_RegisterClientAgent_InputMessage

This is the request for the **RegisterClientAgent** operation.

```
<wsdl:message name="IAlertSynchProviderService_RegisterClientAgent_InputMessage">
  <wsdl:part name="parameters" element="tns:RegisterClientAgent" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/RegisterClientAgent
```

The body of the SOAP message MUST contain the **RegisterClientAgent** element.

#### 3.3.5.17.1.2 IAlertSynchProviderService_RegisterClientAgent_OutputMessage

This is the response for the **RegisterClientAgent** operation.

```
<wsdl:message name="IAlertSynchProviderService_RegisterClientAgent_OutputMessage">
  <wsdl:part name="parameters" element="tns:RegisterClientAgentResponse" />
```

```
        </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
        http://tempuri.org/IAlertSynchProviderService/RegisterClientAgentResponse
```

The body of the SOAP message MUST contain the **RegisterClientAgentResponse** element.

### 3.3.5.17.2    Elements

### 3.3.5.17.2.1  RegisterClientAgent

This element specifies input values for the **RegisterClientAgent** operation.

```
        <xs:element name="RegisterClientAgent">
          <xs:complexType>
            <xs:sequence>
              <xs:element minOccurs="0" name="networkMachineId" nillable="true"  type="xs:string" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
```

### 3.3.5.17.2.2  RegisterClientAgentResponse

This element specifies output values for the **RegisterClientAgent** operation.

```
        <xs:element name="RegisterClientAgentResponse">
          <xs:complexType>
            <xs:sequence />
          </xs:complexType>
        </xs:element>
```

### 3.3.5.18        IAlertSynchProviderService.SynchronizeAlerts

This operation synchronizes the list of **AlertInfo** objects representing alerts raised, cleared, updated, or evaluated on a particular machine in the network to the server.

```
        <wsdl:operation name="SynchronizeAlerts">
          <wsdl:input wsam:Action="http://tempuri.org/IAlertSynchProviderService/SynchronizeAlerts"
        message="tns:IAlertSynchProviderService_SynchronizeAlerts_InputMessage" />
        </wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_SynchronizeAlerts_InputMessage** request message, the protocol server starts to synchronize the list of AlertInfo objects.

### 3.3.5.18.1    Messages

### 3.3.5.18.1.1  IAlertSynchProviderService_SynchronizeAlerts_InputMessage

This is the request for the **SynchronizeAlerts** operation.

```
        <wsdl:message name="IAlertSynchProviderService_SynchronizeAlerts_InputMessage">
          <wsdl:part name="parameters" element="tns:SynchronizeAlerts" />
```

```
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/SynchronizeAlerts
```

The body of the SOAP message MUST contain the **SynchronizeAlerts** element.

### 3.3.5.18.2    Elements

### 3.3.5.18.2.1 SynchronizeAlerts

This element specifies input values for the **SynchronizeAlerts** operation.

```
<xs:element name="SynchronizeAlerts">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="alerts" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q1:ArrayOfAlertInfo" />
      <xs:element minOccurs="0" name="networkMachineId" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="task"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.AlertFramework" type="q2:TaskType" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.19        IAlertSynchProviderService.ClearCallbackCommandQueue

This operation clears all pending command queues for the machine.

```
<wsdl:operation name="ClearCallbackCommandQueue">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertSynchProviderService/ClearCallbackCommandQueue"
message="tns:IAlertSynchProviderService_ClearCallbackCommandQueue_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertSynchProviderService/ClearCallbackCommandQueueResponse"
message="tns:IAlertSynchProviderService_ClearCallbackCommandQueue_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_ClearCallbackCommandQueue_InputMessage**
request message, the protocol server starts to clear all pending command queues for the machine.
Upon successful completion of the action, the server MUST respond with
**IAlertSynchProviderService_ClearCallbackCommandQueue_OutputMessage**. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.19.1    Messages

### 3.3.5.19.1.1 IAlertSynchProviderService_ClearCallbackCommandQueue_InputMessag e

This is the request for the **ClearCallbackCommandQueue** operation.

```
<wsdl:message name="IAlertSynchProviderService_ClearCallbackCommandQueue_InputMessage">
  <wsdl:part name="parameters" element="tns:ClearCallbackCommandQueue" />
```

```
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/ClearCallbackCommandQueue
```

The body of the SOAP message MUST contain the **ClearCallbackCommandQueue** element.

### 3.3.5.19.1.2 IAlertSynchProviderService_ClearCallbackCommandQueue_OutputMessage

This is the response for the **ClearCallbackCommandQueue** operation.

```
<wsdl:message name="IAlertSynchProviderService_ClearCallbackCommandQueue_OutputMessage">
  <wsdl:part name="parameters" element="tns:ClearCallbackCommandQueueResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/ClearCallbackCommandQueueResponse
```

The body of the SOAP message MUST contain the **ClearCallbackCommandQueueResponse** element.

### 3.3.5.19.2 Elements

### 3.3.5.19.2.1 ClearCallbackCommandQueue

This element specifies input values for the **ClearCallbackCommandQueue** operation.

```
<xs:element name="ClearCallbackCommandQueue">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="networkMachineId" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.19.2.2 ClearCallbackCommandQueueResponse

This element specifies output values for the **ClearCallbackCommandQueue** operation.

```
<xs:element name="ClearCallbackCommandQueueResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.3.5.20 IAlertSynchProviderService.GetPlugInInformation

This operation gets information for all installed plugins.

```
<wsdl:operation name="GetPlugInInformation">
```

```
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertSynchProviderService/GetPlugInInformation"
message="tns:IAlertSynchProviderService_GetPlugInInformation_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertSynchProviderService/GetPlugInInformationResponse"
message="tns:IAlertSynchProviderService_GetPlugInInformation_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_GetPlugInInformation_InputMessage** request message, the protocol server starts to get the plugin information. Upon successful completion of the action, the server MUST respond with **IAlertSynchProviderService_GetPlugInInformation_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.20.1    Messages

### 3.3.5.20.1.1  IAlertSynchProviderService_GetPlugInInformation_InputMessage

This is the request for the **GetPlugInInformation** operation.

```
<wsdl:message name="IAlertSynchProviderService_GetPlugInInformation_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPlugInInformation" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/GetPlugInInformation
```

The body of the SOAP message MUST contain the **GetPlugInInformation** element.

### 3.3.5.20.1.2  IAlertSynchProviderService_GetPlugInInformation_OutputMessage

This is the response for the **GetPlugInInformation** operation.

```
<wsdl:message name="IAlertSynchProviderService_GetPlugInInformation_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPlugInInformationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/GetPlugInInformationResponse
```

The body of the SOAP message MUST contain the GetPlugInInformationResponse element.

### 3.3.5.20.2    Elements

### 3.3.5.20.2.1  GetPlugInInformation

This element specifies input values for the **GetPlugInInformation** operation.

```
<xs:element name="GetPlugInInformation">
 <xs:complexType>
   <xs:sequence />
 </xs:complexType>
```

```
  </xs:element>
```

### 3.3.5.20.2.2 GetPlugInInformationResponse

This element specifies output values for the **GetPlugInInformation** operation.

```
<xs:element name="GetPlugInInformationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetPlugInInformationResult" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.NetworkHea
lth.Engine" type="q3:ArrayOfPlugInInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.3.5.21       IAlertSynchProviderService.DownloadDefinition

This operation obtains a stream containing the definition file for a feature plug-in.

```
<wsdl:operation name="DownloadDefinition">
  <wsdl:input wsam:Action="http://tempuri.org/IAlertSynchProviderService/DownloadDefinition"
message="tns:IAlertSynchProviderService_DownloadDefinition_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertSynchProviderService/DownloadDefinitionResponse"
message="tns:IAlertSynchProviderService_DownloadDefinition_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_DownloadDefinition_InputMessage** request message, the protocol server generates a stream containing the definition file for the feature plug-in. Upon successful completion of the action, the server MUST respond with **IAlertSynchProviderService_DownloadDefinition_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.21.1    Messages

#### 3.3.5.21.1.1 IAlertSynchProviderService_DownloadDefinition_InputMessage

This is the request for the **DownloadDefinition** operation.

```
<wsdl:message name="IAlertSynchProviderService_DownloadDefinition_InputMessage">
  <wsdl:part name="parameters" element="tns:DownloadDefinition" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/DownloadDefinition
```

The body of the SOAP message MUST contain the **DownloadDefinition** element.

#### 3.3.5.21.1.2 IAlertSynchProviderService_DownloadDefinition_OutputMessage

This is the response for the **DownloadDefinition** operation.

```
<wsdl:message name="IAlertSynchProviderService_DownloadDefinition_OutputMessage">
```

```
            <wsdl:part name="parameters" element="tns:DownloadDefinitionResponse" />
        </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/DownloadDefinitionResponse
```

The body of the SOAP message MUST contain the **DownloadDefinitionResponse** element.

### 3.3.5.21.2    Elements

#### 3.3.5.21.2.1  DownloadDefinition

This element specifies input values for the **DownloadDefinition** operation.

```
<xs:element name="DownloadDefinition">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="featureName" nillable="true"  type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.3.5.21.2.2  DownloadDefinitionResponse

This element specifies output values for the **DownloadDefinition** operation.

```
<xs:element name="DownloadDefinitionResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="DownloadDefinitionResult"
xmlns:q4="http://schemas.microsoft.com/Message" type="q4:StreamBody" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.22        IAlertSynchProviderService.DownloadConfiguration

This operation obtains a stream containing the definition configuration file for a feature plug-in.

```
<wsdl:operation name="DownloadConfiguration">
  <wsdl:input
wsam:Action="http://tempuri.org/IAlertSynchProviderService/DownloadConfiguration"
message="tns:IAlertSynchProviderService_DownloadConfiguration_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAlertSynchProviderService/DownloadConfigurationResponse"
message="tns:IAlertSynchProviderService_DownloadConfiguration_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IAlertSynchProviderService_DownloadConfiguration_InputMessage**
request message, the protocol server generates a stream containing the definition configuration file
for the feature plug-in. Upon successful completion of the action, the server MUST respond with
**IAlertSynchProviderService_DownloadConfiguration_OutputMessage**. In the event of a failure,
an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.3.5.22.1    Messages

### 3.3.5.22.1.1 IAlertSynchProviderService_DownloadConfiguration_InputMessage

This is the request for the **DownloadConfiguration** operation.

```
<wsdl:message name="IAlertSynchProviderService_DownloadConfiguration_InputMessage">
  <wsdl:part name="parameters" element="tns:DownloadConfiguration" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/DownloadConfiguration
```

The body of the SOAP message MUST contain the **DownloadConfiguration** element.

### 3.3.5.22.1.2 IAlertSynchProviderService_DownloadConfiguration_OutputMessage

This is the response for the **DownloadConfiguration** operation.

```
<wsdl:message name="IAlertSynchProviderService_DownloadConfiguration_OutputMessage">
  <wsdl:part name="parameters" element="tns:DownloadConfigurationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAlertSynchProviderService/DownloadConfigurationResponse
```

The body of the SOAP message MUST contain the **DownloadConfigurationResponse** element.

### 3.3.5.22.2    Elements

### 3.3.5.22.2.1 DownloadConfiguration

This element specifies input values for the **DownloadConfiguration** operation.

```
<xs:element name="DownloadConfiguration">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="featureName" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.5.22.2.2 DownloadConfigurationResponse

This element specifies input values for the **DownloadConfiguration** operation.

```
<xs:element name="DownloadConfigurationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="DownloadConfigurationResult"
xmlns:q5="http://schemas.microsoft.com/Message" type="q5:StreamBody" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.3.6  Timer Events

None.

### 3.3.7  Other Local Events

None.

## 3.4  Machine Identity Provider Service Contract Details

### 3.4.1  Abstract Data Model

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

#### 3.4.1.1  Machine Identity

The protocol server saves certificates issued for each add machine request, and get machine identity status from the system certificates store.

#### 3.4.1.2  Machine RDP Permission

The protocol server maintains the user-computer RDP permission allowance mapping that tells who has permission to connect to which machine using remote desktop. This can be achieved by creating one Activate Directory security group for each machine and then joining the allowed users to the group.

### 3.4.2  Timers

None.

### 3.4.3  Initialization

None.

### 3.4.4  Higher-Layer Triggered Events

None.

### 3.4.5  Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IMachineIdentityProviderCallback.ReturnNewCertResponse | Callback for the New Certificate Response. |
| IMachineIdentityProviderCallback.ReturnRenewCert | Callback for the Renew New Certificate. |
| IMachineIdentityProviderCallback.ReturnSid | Callback for the machine SID. This happens only when a new machine is added to the Colorado system and |

| OperationContract | Description |
|---|---|
| | the machine connects with the server. SID is the ID for this machine. |
| IMachineIdentityProviderCallback.ReturnRevokeCert | Callback for the Revoke Certificate. |
| IMachineIdentityProviderCallback.ReturnRemoveMachine | Callback to remove a machine. This happens after the connection between the server and client terminates. |
| IMachineIdentityProviderCallback.ReturnGetMachineStatus | Get the machine status. |
| IMachineIdentityProviderCallback.ReturnMakeUserRemoteDesktopMapping | Callback for the MakeUserRemoteDesktopMapping operation. |
| IMachineIdentityProvider.RemoveMachine | Remove machine from the managed network. This ~~should be~~is called from the server side. |
| IMachineIdentityProvider.RenewCert | Renew the caller's certificate. |
| IMachineIdentityProvider.RevokeCert | Revoke the caller's certificate. |
| IMachineIdentityProvider.GenerateCertResponse | This function is called by Enroll WebService. UserName/Password required to authenticate the user. |
| IMachineIdentityProvider.GenerateSid | Generate the SID for the machine. UserName/Password required to authenticate the user. |
| IMachineIdentityProvider.GetMachineStatus | Returns the status of machine named by the callback. |
| IMachineIdentityProvider.MakeUserRemoteDesktopMapping | Map a user to a particular remote desktop computer. |

### 3.4.5.1  IMachineIdentityProvider.ReturnNewCertResponse

This operation is the callback for the **IMachineIdentityProvider.GenerateCertResponse** operation.

```
<wsdl:operation name="ReturnNewCertResponse">
  <wsdl:output
wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnNewCertResponse"
message="tns:IMachineIdentityProvider_ReturnNewCertResponse_OutputCallbackMessage" />
</wsdl:operation>
```

#### 3.4.5.1.1 Messages

#### 3.4.5.1.1.1   IMachineIdentityProvider_ReturnNewCertResponse_OutputCallbackMessage

This is the request for the **ReturnNewCertResponse** operation.

```
<wsdl:message name="IMachineIdentityProviderCallback_ReturnNewCertResponse_OutputMessage">
```

```
            <wsdl:part name="parameters" element="tns:ReturnNewCertResponse" />
        </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IMachineIdentityProvider/ReturnNewCertResponse
```

The body of the SOAP message MUST contain the **ReturnNewCertResponse** element.

### 3.4.5.1.2 Elements

#### 3.4.5.1.2.1    ReturnNewCertResponse

This element specifies input values for the **ReturnNewCertResponse** operation. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
    <xs:element name="ReturnNewCertResponse">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" name="newCert" nillable="true" type="xs:string" />
          <xs:element minOccurs="0" name="returnCode" nillable="true"
    xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
    entity" type="q1:ReturnClass" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
```

### 3.4.5.2  IMachineIdentityProvider.ReturnRenewCert

This operation is the callback for the **IMachineIdentityProvider.RenewCert** operation. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
    <wsdl:operation name="ReturnRenewCert">
      <wsdl:output wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnRenewCert"
    message="tns:IMachineIdentityProvider_ReturnRenewCert_OutputCallbackMessage" />
    </wsdl:operation>
```

### 3.4.5.2.1 Messages

#### 3.4.5.2.1.1    IMachineIdentityProvider_ReturnRenewCert_OutputCallbackMessage

This is the request for the **ReturnRenewCert** operation.

```
    <wsdl:message name="IMachineIdentityProvider_ReturnRenewCert_OutputCallbackMessage">
      <wsdl:part name="parameters" element="tns:ReturnRenewCert" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IMachineIdentityProvider/ReturnRenewCert
```

The body of the SOAP message MUST contain the **ReturnRenewCert** element.

### 3.4.5.2.2 Elements

### 3.4.5.2.2.1 ReturnRenewCert

This element specifies input values for the **ReturnRenewCert** operation.

```
<xs:element name="ReturnRenewCert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="newCert" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q2:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.3 IMachineIdentityProvider.ReturnSid

This operation is the callback for the **IMachineIdentityProvider.GenerateSid** operation.

```
<wsdl:operation name="ReturnSid">
  <wsdl:output wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnSid"
message="tns:IMachineIdentityProvider_ReturnSid_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.4.5.3.1 Messages

### 3.4.5.3.1.1 IMachineIdentityProvider_ReturnSid_OutputCallbackMessage

This is the request for the **ReturnSid** operation.

```
<wsdl:message name="IMachineIdentityProvider_ReturnSid_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnSid" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/ReturnSid
```

The body of the SOAP message MUST contain the **ReturnSid** element.

### 3.4.5.3.2 Elements

### 3.4.5.3.2.1 ReturnSid

This element specifies input values for the **ReturnSid** operation.

```
<xs:element name="ReturnSid">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="sid" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q3:ReturnClass" />
    </xs:sequence>
```

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

### 3.4.5.4 IMachineIdentityProvider.ReturnRevokeCert

This operation is the callback for the **IMachineIdentityProvider.RevokeCert** operation.

```
<wsdl:operation name="ReturnRevokeCert">
  <wsdl:output wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnRevokeCert"
message="tns:IMachineIdentityProvider_ReturnRevokeCert_OutputCallbackMessage" />
</wsdl:operation>
```

#### 3.4.5.4.1 Messages

##### 3.4.5.4.1.1    IMachineIdentityProvider_ReturnRevokeCert_OutputCallbackMessage

This is the request for the **ReturnRevokeCert** operation.

```
<wsdl:message name="IMachineIdentityProvider_ReturnRevokeCert_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnRevokeCert" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/ReturnRevokeCert
```

The body of the SOAP message MUST contain the **ReturnRevokeCert** element.

#### 3.4.5.4.2 Elements

##### 3.4.5.4.2.1    ReturnRevokeCert

This element specifies input values for the **ReturnRevokeCert** operation.

```
<xs:element name="ReturnRevokeCert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q4:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.5 IMachineIdentityProvider.ReturnRemoveMachine

This operation is the callback for the **IMachineIdentityProvider.RemoveMachine** operation<13>.

```
<wsdl:operation name="ReturnRemoveMachine">
  <wsdl:output wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnRemoveMachine"
message="tns:IMachineIdentityProvider_ReturnRemoveMachine_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.4.5.5.1 Messages

### 3.4.5.5.1.1 IMachineIdentityProvider_ReturnRemoveMachine_OutputCallbackMessage

This is the request for the **ReturnRemoveMachine** operation.

```
<wsdl:message name="IMachineIdentityProvider_ReturnRemoveMachine_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnRemoveMachine" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/ReturnRemoveMachine
```

The body of the SOAP message MUST contain the **ReturnRemoveMachine** element.

### 3.4.5.5.2 Elements

### 3.4.5.5.2.1 ReturnRemoveMachine

This element specifies input values for the **ReturnRemoveMachine** operation.

```
<xs:element name="ReturnRemoveMachine">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q5:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.6 IMachineIdentityProvider.ReturnGetMachineStatus

This operation is the callback for the **IMachineIdentityProvider.GetMachineStatus** operation.

```
<wsdl:operation name="ReturnGetMachineStatus">
  <wsdl:output
wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnGetMachineStatus"
message="tns:IMachineIdentityProvider_ReturnGetMachineStatus_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.4.5.6.1 Messages

### 3.4.5.6.1.1 IMachineIdentityProvider_ReturnGetMachineStatus_OutputCallbackMessage

This is the request for the **ReturnGetMachineStatus** operation.

```
<wsdl:message name="IMachineIdentityProvider_ReturnGetMachineStatus_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnGetMachineStatus" />
</wsdl:message>
```

```
This message MUST be sent with the following SOAP action.
http://tempuri.org/IMachineIdentityProvider/ReturnGetMachineStatus
```

The body of the SOAP message MUST contain the **ReturnGetMachineStatus** element.

### 3.4.5.6.2 Elements

#### 3.4.5.6.2.1   ReturnGetMachineStatus

This element specifies input values for the **ReturnGetMachineStatus** operation.

```
<xs:element name="ReturnGetMachineStatus">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="status" nillable="true"
xmlns:q6="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q6:GetMachineStatusType" />
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q7="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q7:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.7 IMachineIdentityProvider.ReturnMakeUserRemoteDesktopMapping

This operation is the callback for the
**IMachineIdentityProvider.MakeUserRemoteDesktopMapping** operation.

```
<wsdl:operation name="ReturnMakeUserRemoteDesktopMapping">
  <wsdl:output
wsam:Action="http://tempuri.org/IMachineIdentityProvider/ReturnMakeUserRemoteDesktopMapping"
message="tns:IMachineIdentityProvider_ReturnMakeUserRemoteDesktopMapping_OutputCallbackMessag
e" />
</wsdl:operation>
```

### 3.4.5.7.1 Messages

#### 3.4.5.7.1.1   IMachineIdentityProvider_ReturnMakeUserRemoteDesktopMapping_Out putCallbackMessage

This is the request for the **ReturnMakeUserRemoteDesktopMapping** operation.

```
<wsdl:message
name="IMachineIdentityProvider_ReturnMakeUserRemoteDesktopMapping_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnMakeUserRemoteDesktopMapping" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/ReturnMakeUserRemoteDesktopMapping
```

The body of the SOAP message MUST contain the **ReturnMakeUserRemoteDesktopMapping**
element.

### 3.4.5.7.2 Elements

### 3.4.5.7.2.1 ReturnMakeUserRemoteDesktopMapping

This element specifies input values for the **ReturnMakeUserRemoteDesktopMapping** operation.

```
<xs:element name="ReturnMakeUserRemoteDesktopMapping">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="returnCode" nillable="true"
xmlns:q8="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Devices.Id
entity" type="q8:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.8 IMachineIdentityProvider.RemoveMachine

This operation removes a machine from the managed network.<14>

```
<wsdl:operation name="RemoveMachine">
  <wsdl:input wsam:Action="http://tempuri.org/IMachineIdentityProvider/RemoveMachine"
message="tns:IMachineIdentityProvider_RemoveMachine_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_RemoveMachine_InputMessage** request
message, the protocol server is notified to remove the specific machine. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.4.5.8.1 Messages

### 3.4.5.8.1.1 IMachineIdentityProvider_RemoveMachine_InputMessage

This is the request for the **RemoveMachine** operation.

```
<wsdl:message name="IMachineIdentityProvider_RemoveMachine_InputMessage">
  <wsdl:part name="parameters" element="tns:RemoveMachine" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/RemoveMachine
```

The body of the SOAP message MUST contain the **RemoveMachine** element.

### 3.4.5.8.2 Elements

### 3.4.5.8.2.1 RemoveMachine

This element specifies input values for the **RemoveMachine** operation.

```
<xs:element name="RemoveMachine">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="machineID" type="xs:string" />
      <xs:element minOccurs="0" name="serialNo" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.9  IMachineIdentityProvider.RenewCert

This operation renews the caller's certificate.

```
<wsdl:operation name="RenewCert">
  <wsdl:input wsam:Action="http://tempuri.org/IMachineIdentityProvider/RenewCert"
message="tns:IMachineIdentityProvider_RenewCert_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_RenewCert_InputMessage** request message, the protocol server is notified to renew the caller's certificate. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.4.5.9.1 Messages

##### 3.4.5.9.1.1   IMachineIdentityProvider_RenewCert_InputMessage

This is the request for the **RenewCert** operation.

```
<wsdl:message name="IMachineIdentityProvider_RenewCert_InputMessage">
  <wsdl:part name="parameters" element="tns:RenewCert" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/RenewCert
```

The body of the SOAP message MUST contain the **RenewCert** element.

#### 3.4.5.9.2 Elements

##### 3.4.5.9.2.1   RenewCert

This element specifies input values for the **RenewCert** operation.

```
<xs:element name="RenewCert">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="strCSR" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.10      IMachineIdentityProvider.RevokeCert

This operation revokes the caller's certificate.

```
<wsdl:operation name="RevokeCert">
```

```
    <wsdl:output wsaw:Action="net.tcp://Microsoft.WindowsServerSolutions.Devices.Identity/
    IMachineIdentityProvider/RevokeCertResponse"
    message="IMachineIdentityProvider_RevokeCert_InputMessage" />
  </wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_RevokeCert_InputMessage** request message, the protocol server is notified to revoke the caller's certificate. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.4.5.10.1    Messages

### 3.4.5.10.1.1 IMachineIdentityProvider_RevokeCert_InputMessage

This is the response for the **RevokeCert** operation.

```
    <wsdl:message name="IMachineIdentityProvider_RevokeCert_InputMessage">
      <wsdl:part name="parameters" element="tns:RevokeCert" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IMachineIdentityProvider/RevokeCert
```

The body of the SOAP message MUST contain the **RevokeCertResponse** element.

### 3.4.5.10.2    Elements

### 3.4.5.10.2.1 RevokeCert

This element specifies input values for the **RevokeCert** operation.

```
    <xs:element name="RevokeCert">
      <xs:complexType>
        <xs:sequence />
      </xs:complexType>
    </xs:element>
```

### 3.4.5.11        IMachineIdentityProvider.GenerateCertResponse

This operation generates a certificate for a machine.

```
    <wsdl:operation name="GenerateCertResponse">
      <wsdl:input wsam:Action="http://tempuri.org/IMachineIdentityProvider/GenerateCertResponse"
    message="tns:IMachineIdentityProvider_GenerateCertResponse_InputMessage" />
    </wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_GenerateCertResponse_InputMessage** request message, the protocol server is notified to generate a certificate for the machine. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.4.5.11.1    Messages

### 3.4.5.11.1.1 IMachineIdentityProvider_GenerateCertResponse_InputMessage

This is the request for the **GenerateCertResponse** operation.

```
<wsdl:message name="IMachineIdentityProvider_GenerateCertResponse_InputMessage">
  <wsdl:part name="parameters" element="tns:GenerateCertResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/GenerateCertResponse
```

The body of the SOAP message MUST contain the **GenerateCertResponse** element.

### 3.4.5.11.2    Elements

### 3.4.5.11.2.1  GenerateCertResponse

This element specifies input values for the **GenerateCertResponse** operation.

```
<xs:element name="GenerateCertResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" type="xs:string" />
      <xs:element minOccurs="0" name="password" type="xs:string" />
      <xs:element minOccurs="0" name="strCSR" type="xs:string" />
      <xs:element minOccurs="0" name="machineName" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.12      IMachineIdentityProvider.GenerateSid

This operation generates the SID for a machine.

```
<wsdl:operation name="GenerateSid">
  <wsdl:input wsam:Action="http://tempuri.org/IMachineIdentityProvider/GenerateSid"
message="tns:IMachineIdentityProvider_GenerateSid_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_GenerateSid_InputMessage** request message, the
protocol server is notified to generate the SID for the machine. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.4.5.12.1    Messages

### 3.4.5.12.1.1  IMachineIdentityProvider_GenerateSid_InputMessage

This is the request for the **GenerateSid** operation.

```
<wsdl:message name="IMachineIdentityProvider_GenerateSid_InputMessage">
  <wsdl:part name="parameters" element="tns:GenerateSid" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/GenerateSid
```

The body of the SOAP message MUST contain the **GenerateSid** element.

### 3.4.5.12.2    Elements

#### 3.4.5.12.2.1  GenerateSid

This element specifies input values for the **GenerateSid** operation.

```
<xs:element name="GenerateSid">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" type="xs:string" />
      <xs:element minOccurs="0" name="password" type="xs:string" />
      <xs:element minOccurs="0" name="machineName" type="xs:string" />
      <xs:element minOccurs="0" name="bForce" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.13        IMachineIdentityProvider.GetMachineStatus

This operation returns the status of the machine named by the callback.

```
<wsdl:operation name="GetMachineStatus">
  <wsdl:input wsam:Action="http://tempuri.org/IMachineIdentityProvider/GetMachineStatus"
message="tns:IMachineIdentityProvider_GetMachineStatus_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IMachineIdentityProvider_GetMachineStatus_InputMessage** request
message, the protocol server is notified to return the status of the machine. In the event of a failure,
an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.4.5.13.1    Messages

#### 3.4.5.13.1.1  IMachineIdentityProvider_GetMachineStatus_InputMessage

This is the request for the **GetMachineStatus** operation.

```
<wsdl:message name="IMachineIdentityProvider_GetMachineStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:GetMachineStatus" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/GetMachineStatus
```

The body of the SOAP message MUST contain the **GetMachineStatus** element.

### 3.4.5.13.2    Elements

#### 3.4.5.13.2.1  GetMachineStatus

This element specifies input values for the **GetMachineStatus** operation.

```
<xs:element name="GetMachineStatus">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" type="xs:string" />
      <xs:element minOccurs="0" name="password" type="xs:string" />
      <xs:element minOccurs="0" name="machineName" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.4.5.14    IMachineIdentityProvider.MakeUserRemoteDesktopMapping

This operation maps a user to a particular computer.

```
<wsdl:operation name="MakeUserRemoteDesktopMapping">
  <wsdl:input
wsam:Action="http://tempuri.org/IMachineIdentityProvider/MakeUserRemoteDesktopMapping"
message="tns:IMachineIdentityProvider_MakeUserRemoteDesktopMapping_InputMessage" />
</wsdl:operation>
```

Upon receiving the
**IMachineIdentityProvider_MakeUserRemoteDesktopMapping_InputMessage** request
message, the protocol server is notified to map a user to a particular computer. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.4.5.14.1    Messages

#### 3.4.5.14.1.1  IMachineIdentityProvider_MakeUserRemoteDesktopMapping_InputMess age

This is the request for the **MakeUserRemoteDesktopMapping** operation.

```
<wsdl:message name="IMachineIdentityProvider_MakeUserRemoteDesktopMapping_InputMessage">
  <wsdl:part name="parameters" element="tns:MakeUserRemoteDesktopMapping" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IMachineIdentityProvider/MakeUserRemoteDesktopMapping
```

The body of the SOAP message MUST contain the **MakeUserRemoteDesktopMapping** element.

#### 3.4.5.14.2    Elements

#### 3.4.5.14.2.1  MakeUserRemoteDesktopMapping

This element specifies input values for the **MakeUserRemoteDesktopMapping** operation.

```
<xs:element name="MakeUserRemoteDesktopMapping">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" type="xs:string" />
    </xs:sequence>
```

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

## 3.4.6  Timer Events

None.

## 3.4.7  Other Local Events

None.

## 3.5  User Identity Provider Service Contract Details

### 3.5.1  Abstract Data Model

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

#### 3.5.1.1  ADM_KeyValueOfUser

This is a dictionary of user information for all domain users.

**Key**: The SID of the user.

**Value**: Information about the given user.

#### 3.5.1.1.1 Data Model

##### 3.5.1.1.1.1    UserInfo

Note that the UserInfo data model also has a corresponding SOAP complex type defined in section 2.2.4.5.2.

**FullUserName:** The domain/ServerName of the user. Mainly used in Launchpad.

**PersonalInfo:** Detailed personal information about the given user. The PersonalInfo type is defined in section 2.2.4.5.1; see also the details in section 3.5.1.1.1.1.1.

**UserName:** The name of the user.

**UserStatus:** The status of the user. The UserStatus type is defined in section 2.2.5.5.5.

**Warning:** The last error of import/update/add user.

**WindowsAccount:** The account of this user.<15>

**WindowsSID:** The SID of this user.<16>

##### 3.5.1.1.1.1.1 PersonalInfo

**AccessLevel**: The access level of the user. The AccessLevelType is defined in section 2.2.5.5.4.

**FirstName**: The first name of the user.

**IsBuiltIn**: TRUE if the user is built-in; otherwise FALSE.

**IsLocalUser**: TRUE if the user is local; otherwise FALSE.

**IsShowInUI**: TRUE if this user ~~should show~~shows in the UI; otherwise FALSE.

**LastName**: The last name of the user.

**RemoteAccess**: The remote access status of the user. RemoteAccessType is defined in section 2.2.5.5.3.

**UserFolder**: The path of the shared folder belonging to this user.

**VpnAccess**: The VPN access status of the user. VPNAccessType is defined in section 2.2.5.5.3.

### 3.5.2  Timers

None.

### 3.5.3  Initialization

None.

### 3.5.4  Higher-Layer Triggered Events

None.

### 3.5.5  Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IUserIdentityProvider.ChangePassword | Change the user password. |
| IUserLogonProvider.LogonUser | Log on a user (who provides the correct password) and display the requested UserInfo. |
| IUserChangePasswordProvider.ChangePassword | Change the user password. |
| IUserInfoProvider.ReturnGroups | Callback to OM after GetGroups. |
| IUserInfoProvider.ReturnWSSServerGroups | Callback to OM after GetWSSServerGroups. |
| IUserInfoProvider.ReturnIsPartOfServerUsers | Callback to OM after IsPartOfServerUsers. |
| IUserInfoProvider.GetGroups | Get groups for the current user. |
| IUserInfoProvider.GetWSSServerGroups | Get WSSServer Groups for the current user. |
| IUserInfoProvider.IsPartOfServerUsers | Check whether the use is server users. |

### 3.5.5.1  IUserLogonProvider.LogonUser

This operation logs a user on and displays the requested UserInfo. <17>

```
<wsdl:operation name="LogonUser">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
```

```
3/2010/IUserLogonProvider/LogonUser" message="tns:IUserLogonProvider_LogonUser_InputMessage"
/>
  <wsdl:output
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserLogonProvider/LogonUserResponse"
message="tns:IUserLogonProvider_LogonUser_OutputMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserLogonProvider/LogonUserUserFaultFault" name="UserFaultFault"
message="tns:IUserLogonProvider_LogonUser_UserFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserLogonProvider/LogonUserUserExceptionFaultFault" name="UserExceptionFaultFault"
message="tns:IUserLogonProvider_LogonUser_UserExceptionFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserLogonProvider/LogonUserUserExceptionFault" name="UserExceptionFault"
message="tns:IUserLogonProvider_LogonUser_UserExceptionFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IUserLogonProvider_LogonUser_InputMessage** request message, the
protocol server logs on the user if they provided the correct password. Upon successful completion of
the action, the server MUST respond with **IUserLogonProvider_LogonUser_OutputMessage**,
which includes the user information object. In the event of a failure, an appropriate SOAP fault MUST
be sent to the client as specified in section 2.2.2.1.

### 3.5.5.1.1 Messages

#### 3.5.5.1.1.1   IUserLogonProvider_LogonUser_InputMessage

This is the request for the **LogonUser** operation.

```
<wsdl:message name="IUserLogonProvider_LogonUser_InputMessage">
  <wsdl:part name="parameters" element="tns:LogonUser" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserL
ogonProvider/LogonUser
```

The body of the SOAP message MUST contain the **LogonUser** element.

#### 3.5.5.1.1.2   IUserLogonProvider_LogonUser_OutputMessage

This is the response for the **LogonUser** operation.

```
<wsdl:message name="IUserLogonProvider_LogonUser_OutputMessage">
  <wsdl:part name="parameters" element="tns:LogonUserResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
dentityProvider/LogonUserResponse
```

The body of the SOAP message MUST contain the **LogonUserResponse** element.

### 3.5.5.1.2 Elements

#### 3.5.5.1.2.1 LogonUser

This element specifies input values for the **LogonUser** operation.

```
<xs:element name="LogonUser">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" nillable="true"  type="xs:string" />
      <xs:element minOccurs="0" name="password" nillable="true"  type="xs:string" />
      <xs:element minOccurs="0" name="bCleanSession" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.5.5.1.2.2 LogonUserResponse

This element specifies output values for the **LogonUser** operation.

```
<xs:element name="LogonUserResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="LogonUserResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Users"
type="q1:UserInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.5.5.2 IUserChangePasswordProvider.ChangePassword

This operation changes a user's password. <18>

```
<wsdl:operation name="ChangePassword">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/ IUserChangePasswordProvider /ChangePassword" message="tns:
IUserChangePasswordProvider _ChangePassword_InputMessage" />
  <wsdl:output
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/ IUserChangePasswordProvider /ChangePasswordResponse" message="tns:
IUserChangePasswordProvider _ChangePassword_OutputMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/ IUserChangePasswordProvider /ChangePasswordUserExceptionFault"
name="UserExceptionFault" message="tns: IUserChangePasswordProvider
_ChangePassword_UserExceptionFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/ IUserChangePasswordProvider /ChangePasswordUserExceptionFaultFault"
name="UserExceptionFaultFault" message="tns: IUserChangePasswordProvider
_ChangePassword_UserExceptionFaultFault_FaultMessage" />
  <wsdl:fault
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/ IUserChangePasswordProvider /ChangePasswordUserFaultFault" name="UserFaultFault"
message="tns: IUserChangePasswordProvider _ChangePassword_UserFaultFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IUserChangePasswordProvider _ChangePassword_InputMessage** request message, the protocol server changes the specified user's password. Upon successful completion of the action, the server MUST respond with **IUserChangePasswordProvider _ChangePassword_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.5.5.2.1 Messages

#### 3.5.5.2.1.1    IUserChangePasswordProvider_ChangePassword_InputMessage

This is the request for the **ChangePassword** operation.

```
<wsdl:message name="IUserChangePasswordProvider_ChangePassword_InputMessage">
  <wsdl:part name="parameters" element="tns:ChangePassword" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserC
hangePasswordProvider/ChangePassword
```

The body of the SOAP message MUST contain the **ChangePassword** element.

#### 3.5.5.2.1.2    IUserChangePasswordProvider_ChangePassword_OutputMessage

This is the response for the **ChangePassword** operation.

```
<wsdl:message name="IUserChangePasswordProvider_ChangePassword_OutputMessage">
  <wsdl:part name="parameters" element="tns:ChangePasswordResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserC
hangePasswordProvider/ChangePasswordResponse
```

The body of the SOAP message MUST contain the **ChangePasswordResponse** element.

### 3.5.5.2.2 Elements

#### 3.5.5.2.2.1    ChangePassword

This element specifies input values for the **ChangePassword** operation.

```
<xs:element name="ChangePassword">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="userName" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="oldPassword" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="newPassword" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.5.5.2.2.2 ChangePasswordResponse

This element specifies output values for the **ChangePassword** operation.

```
<xs:element name="ChangePasswordResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="1" name="ChangePasswordResult" type="UserInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.5.5.3 IUserInfoProvider.ReturnGroups

This operation is a callback for the **IUserInfoProvider.GetGroups** operation. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
<wsdl:operation name="ReturnGroups">
  <wsdl:output
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/ReturnGroups"
  message="tns:IUserInfoProvider_ReturnGroups_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_ReturnGroups_OutputCallbackMessage** request message, the protocol client performs the following processing steps. In the event of a failure, an appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

#### 3.5.5.3.1 Messages

##### 3.5.5.3.1.1 IUserInfoProvider_ReturnGroups_OutputCallbackMessage

This is the request for the **ReturnGroups** operation.

```
<wsdl:message name="IUserInfoProviderCallback_ReturnGroups_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnGroups" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
nfoProviderCallback/ReturnGroups
```

The body of the SOAP message MUST contain the **ReturnGroups** element.

#### 3.5.5.3.2 Elements

##### 3.5.5.3.2.1 ReturnGroups

This element specifies input values for the **ReturnGroups** operation.

```
<xs:element name="ReturnGroups">
  <xs:complexType>
    <xs:sequence>
```

```
        <xs:element minOccurs="0" name="groups" nillable="true"
xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q1:ArrayOfstring"
/>
        <xs:element minOccurs="0" name="ret" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Users"
type="q2:ReturnClass" />
      </xs:sequence>
    </xs:complexType>
</xs:element>
```

### 3.5.5.4 IUserInfoProvider.ReturnWSSServerGroups

This operation is callback for the **IUserInfoProvider.GetWSSServerGroups** operation. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
<wsdl:operation name="ReturnWSSServerGroups">
  <wsdl:output
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/ReturnWSSServerGroups"
message="tns:IUserInfoProvider_ReturnWSSServerGroups_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_ReturnWSSServerGroups_OutputCallbackMessage** request message, the protocol client performs the following processing steps. In the event of a failure, an appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

### 3.5.5.4.1 Messages

#### 3.5.5.4.1.1    IUserInfoProvider_ReturnWSSServerGroups_OutputCallbackMessage

This is the request for the **ReturnWSSServerGroups** operation.

```
<wsdl:message name="IUserInfoProviderCallback_ReturnWSSServerGroups_OutputCallbackMessage">
  <wsdl:part name="parameters" element="ReturnWSSServerGroups" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://Microsoft.WindowsServerSolutions.Users/IUserInfoProviderCallback/ReturnWSSServerGroups
```

The body of the SOAP message MUST contain the **ReturnWSSServerGroups** element.

### 3.5.5.4.2 Elements

#### 3.5.5.4.2.1    ReturnWSSServerGroups

This element specifies input values for the **ReturnWSSServerGroups** operation.

```
<xs:element name="ReturnWSSServerGroups">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="groups" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Users"
type="q3:ArrayOfWSSServerGroupType" />
      <xs:element minOccurs="0" name="ret" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Users"
type="q4:ReturnClass" />
```

```
          </xs:sequence>
      </xs:complexType>
   </xs:element>
```

### 3.5.5.5 IUserInfoProvider.ReturnIsPartOfServerUsers

This operation is a callback for the **IUserInfoProvider.IsPartOfServerUsers** operation. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
<wsdl:operation name="ReturnIsPartOfServerUsers">
  <wsdl:output
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/ReturnIsPartOfServerUsers"
  message="tns:IUserInfoProvider_ReturnIsPartOfServerUsers_OutputCallbackMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_ReturnIsPartOfServerUsers_OutputCallbackMessage** request message, the protocol client performs the following processing steps. In the event of a failure, an appropriate SOAP fault MUST be sent to the server as specified in section 2.2.2.1.

#### 3.5.5.5.1 Messages

##### 3.5.5.5.1.1    IUserInfoProvider_ReturnIsPartOfServerUsers_OutputCallbackMessage

This is the request for the **ReturnIsPartOfServerUsers** operation.

```
<wsdl:message
name="IUserInfoProviderCallback_ReturnIsPartOfServerUsers_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ReturnIsPartOfServerUsers" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
nfoProviderCallback/ReturnIsPartOfServerUsers
```

The body of the SOAP message MUST contain the **ReturnIsPartOfServerUsers** element.

#### 3.5.5.5.2 Elements

##### 3.5.5.5.2.1    ReturnIsPartOfServerUsers

This element specifies input values for the **ReturnIsPartOfServerUsers** operation.

```
<xs:element name="ReturnIsPartOfServerUsers">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="bIsPartOfServerUsers" type="xs:boolean" />
      <xs:element minOccurs="0" name="ret" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Users"
type="q5:ReturnClass" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.5.5.6 IUserInfoProvider.GetGroups

This operation returns the groups for the current user.

```
<wsdl:operation name="GetGroups">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/GetGroups" message="tns:IUserInfoProvider_GetGroups_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_GetGroups_InputMessage** request message, the protocol server starts to get the groups for current user. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.5.5.6.1 Messages

#### 3.5.5.6.1.1   IUserInfoProvider_GetGroups_InputMessage

This is the request for the **GetGroups** operation.

```
<wsdl:message name="IUserInfoProvider_GetGroups_InputMessage">
  <wsdl:part name="parameters" element="tns:GetGroups" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
nfoProvider/GetGroups
```

The body of the SOAP message MUST contain the **GetGroups** element.

#### 3.5.5.6.2 Elements

#### 3.5.5.6.2.1   GetGroups

This element specifies input values for the **GetGroups** operation.

```
<xs:element name="GetGroups">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.5.5.7 IUserInfoProvider.GetWSSServerGroups

This operation returns the WSSServer groups for the current user.

```
<wsdl:operation name="GetWSSServerGroups">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/GetWSSServerGroups"
message="tns:IUserInfoProvider_GetWSSServerGroups_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_GetWSSServerGroups_InputMessage** request message, the protocol server starts to get the WSSServer Groups for current user. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.5.5.7.1 Messages

#### 3.5.5.7.1.1   IUserInfoProvider_GetWSSServerGroups_InputMessage

This is the request for the **GetWSSServerGroups** operation.

```
<wsdl:message name="IUserInfoProvider_GetWSSServerGroups_InputMessage">
  <wsdl:part name="parameters" element="tns:GetWSSServerGroups" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
nfoProvider/GetWSSServerGroups
```

The body of the SOAP message MUST contain the **GetWSSServerGroups** element.

### 3.5.5.7.2 Elements

#### 3.5.5.7.2.1   GetWSSServerGroups

This element specifies input values for the **GetWSSServerGroups** operation.

```
<xs:element name="GetWSSServerGroups">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.5.5.8 IUserInfoProvider.IsPartOfServerUsers

This operation returns a value indicating whether the user is a member of the set of server users.

```
<wsdl:operation name="IsPartOfServerUsers">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/IUserInfoProvider/IsPartOfServerUsers"
  message="tns:IUserInfoProvider_IsPartOfServerUsers_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IUserInfoProvider_IsPartOfServerUsers_InputMessage** request message, the protocol server starts to get a Boolean indicating whether the current user is part of server users. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.5.5.8.1 Messages

#### 3.5.5.8.1.1   IUserInfoProvider_IsPartOfServerUsers_InputMessage

This is the request for the **IsPartOfServerUsers** operation.

```
<wsdl:message name="IUserInfoProvider_IsPartOfServerUsers_InputMessage">
  <wsdl:part name="parameters" element="tns:IsPartOfServerUsers" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/IUserI
nfoProvider/IsPartOfServerUsers
```

The body of the SOAP message MUST contain the **IsPartOfServerUsers** element.

### 3.5.5.8.2 Elements

#### 3.5.5.8.2.1    IsPartOfServerUsers

This element specifies input values for the **IsPartOfServerUsers** operation.

```
<xs:element name="IsPartOfServerUsers">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.5.6   Timer Events

None.

### 3.5.7   Other Local Events

None.

## 3.6    Provider Registry Service Contract Details

### 3.6.1   Abstract Data Model

None.

### 3.6.2   Timers

None.

### 3.6.3   Initialization

None.

### 3.6.4   Higher-Layer Triggered Events

None.

### 3.6.5 Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IProviderRegistry.Disconnect | Signal the **ProviderRegistryService** that the connection is terminating. |
| IProviderRegistry.Query | Return a list of **ProviderInfo** data for a given contract. |
| IProviderRegistry.QueryAll | Get all registered providers. |
| IProviderRegistry.RequestProviderInfoUpdate | Request an update for specific providers. |
| IProviderRegistry.GetVersion | Get the provider version. |
| IProviderRegistry.InformOfProviderUpdate | Send an updated **ProviderInfo** object to the connected client. |
| INotificationProviderServiceAdminOnly.Connect | Connect to the notification provider. |
| INotificationProviderServiceAdminOnly.RaiseNotifications | Send persistent notifications to be raised on the provider. |
| INotificationProviderServiceAdminOnly.ClearNotifications | Send persistent notifications to be cleared on the provider. |
| INotificationProviderServiceAdminOnly.RequestNotificationChange | Request a change to the specified persistent notification. |
| INotificationProviderServiceAdminOnly.FireEvent | Send a one-time notification to be raised on the provider. |
| INotificationProviderServiceAdminOnly.AddCategoryFilters | Indicate that the caller is interested in additional categories of notifications. |
| INotificationProviderServiceAdminOnly.RemoveCategoryFilters | Indicate that the caller is no longer interested in the specified categories. |
| INotificationProviderServiceAdminOnly.Disconnect | Disconnect from the notification provider. |

### 3.6.5.1 IProviderRegistry.Disconnect

This operation signals the **ProviderRegistryService** that the connection is terminating.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="true" name="Disconnect">
  <wsdl:input wsam:Action="http://tempuri.org/IProviderRegistryConnection/Disconnect"
message="tns:IProviderRegistry_Disconnect_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IProviderRegistry_Disconnect_InputMessage** request message, the protocol server is notified that the connection is terminating. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.1.1 Messages

### 3.6.5.1.1.1 IProviderRegistry_Disconnect_InputMessage

This is the request for the **Disconnect** operation.

```
<wsdl:message name="IProviderRegistry_Disconnect_InputMessage">
```

```
<wsdl:part name="parameters" element="tns:Disconnect" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistryConnection/Disconnect
```

The body of the SOAP message MUST contain the **Disconnect** element.

### 3.6.5.1.2 Elements

#### 3.6.5.1.2.1   Disconnect

This element specifies input values for the **Disconnect** operation.

```
<xs:element name="Disconnect">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.6.5.2  IProviderRegistry.Query

This operation returns a list of **ProviderInfo** data for a given contract.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="Query">
  <wsdl:input wsam:Action="http://tempuri.org/IProviderRegistry/Query"
message="tns:IProviderRegistry_Query_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IProviderRegistry/QueryResponse"
message="tns:IProviderRegistry_Query_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IProviderRegistry_Query_InputMessage** request message, the protocol server returns a list of **ProviderInfo** data for a given contract. Upon successful completion of the action, the server MUST respond with **IProviderRegistry_Query_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.2.1 Messages

#### 3.6.5.2.1.1   IProviderRegistry_Query_InputMessage

This is the request for the **Query** operation.

```
<wsdl:message name="IProviderRegistry_Query_InputMessage">
  <wsdl:part name="parameters" element="tns:Query" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/Query
```

The body of the SOAP message MUST contain the **Query** element.

### 3.6.5.2.1.2    IProviderRegistry_Query_OutputMessage

This is the response for the **Query** operation.

```
<wsdl:message name="IProviderRegistry_Query_OutputMessage">
  <wsdl:part name="parameters" element="tns:QueryResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/QueryResponse
```

The body of the SOAP message MUST contain the **QueryResponse** element.

### 3.6.5.2.2 Elements

### 3.6.5.2.2.1    Query

This element specifies input values for the **Query** operation.

```
<xs:element name="Query">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="specification" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.Common.Pro
viderFramework" type="q1:QuerySpecification" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.2.2.2    QueryResponse

This element specifies output values for the **Query** operation.

```
<xs:element name="QueryResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="QueryResult" nillable="true"
xmlns:q2="http://www.microsoft.com/2010/06/Microsoft.WindowsServerSolutions.Common.ProviderFr
amework" type="q2:ArrayOfProviderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.3  IProviderRegistry.QueryAll

This operation returns all registered providers.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="QueryAll">
  <wsdl:input wsam:Action="http://tempuri.org/IProviderRegistry/QueryAll"
message="tns:IProviderRegistry_QueryAll_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IProviderRegistry/QueryAllResponse"
message="tns:IProviderRegistry_QueryAll_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IProviderRegistry_QueryAll_InputMessage** request message, the protocol server returns the list of data for all providers. Upon successful completion of the action, the server MUST respond with **IProviderRegistry_QueryAll_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.3.1 Messages

#### 3.6.5.3.1.1    IProviderRegistry_QueryAll_InputMessage

This is the request for the **QueryAll** operation.

```
<wsdl:message name="IProviderRegistry_QueryAll_InputMessage">
  <wsdl:part name="parameters" element="tns:QueryAll" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/QueryAll
```

The body of the SOAP message MUST contain the **QueryAll** element.

#### 3.6.5.3.1.2    IProviderRegistry_QueryAll_OutputMessage

This is the response for the **QueryAll** operation.

```
<wsdl:message name="IProviderRegistry_QueryAll_OutputMessage">
  <wsdl:part name="parameters" element="tns:QueryAllResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/QueryAllResponse
```

The body of the SOAP message MUST contain the **QueryAllResponse** element.

### 3.6.5.3.2 Elements

#### 3.6.5.3.2.1    QueryAll

This element specifies input values for the **QueryAll** operation.

```
<xs:element name="QueryAll">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.6.5.3.2.2    QueryAllResponse

This element specifies output values for the **QueryAll** operation.

```
<xs:element name="QueryAllResponse">
  <xs:complexType>
```

```
    <xs:sequence>
      <xs:element minOccurs="0" name="QueryAllResult" nillable="true"
xmlns:q3="http://www.microsoft.com/2010/06/Microsoft.WindowsServerSolutions.Common.ProviderFr
amework" type="q3:ArrayOfProviderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.6.5.4 IProviderRegistry.RequestProviderInfoUpdate

This operation requests an update for specific providers.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false"
name="RequestProviderInfoUpdate">
  <wsdl:input wsam:Action="http://tempuri.org/IProviderRegistry/RequestProviderInfoUpdate"
message="tns:IProviderRegistry_RequestProviderInfoUpdate_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IProviderRegistry/RequestProviderInfoUpdateResponse"
message="tns:IProviderRegistry_RequestProviderInfoUpdate_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IProviderRegistry_RequestProviderInfoUpdate_InputMessage** request
message, the protocol server performs an update for specific providers. Upon successful completion of
the action, the server MUST respond with
**IProviderRegistry_RequestProviderInfoUpdate_OutputMessage**. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.4.1 Messages

#### 3.6.5.4.1.1    IProviderRegistry_RequestProviderInfoUpdate_InputMessage

This is the request for the **RequestProviderInfoUpdate** operation.

```
<wsdl:message name="IProviderRegistry_RequestProviderInfoUpdate_InputMessage">
  <wsdl:part name="parameters" element="tns:RequestProviderInfoUpdate" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/RequestProviderInfoUpdate
```

The body of the SOAP message MUST contain the **RequestProviderInfoUpdate** element.

#### 3.6.5.4.1.2    IProviderRegistry_RequestProviderInfoUpdate_OutputMessage

This is the response for the **RequestProviderInfoUpdate** operation.

```
<wsdl:message name="IProviderRegistry_RequestProviderInfoUpdate_OutputMessage">
  <wsdl:part name="parameters" element="tns:RequestProviderInfoUpdateResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/RequestProviderInfoUpdateResponse
```

The body of the SOAP message MUST contain the **RequestProviderInfoUpdateResponse** element.

### 3.6.5.4.2 Elements

### 3.6.5.4.2.1 RequestProviderInfoUpdate

This element specifies input values for the **RequestProviderInfoUpdate** operation.

```
<xs:element name="RequestProviderInfoUpdate">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="providersToUpdate" nillable="true"
xmlns:q4="http://www.microsoft.com/2010/06/Microsoft.WindowsServerSolutions.Common.ProviderFr
amework" type="q4:ArrayOfProviderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.4.2.2 RequestProviderInfoUpdateResponse

This element specifies output values for the **RequestProviderInfoUpdate** operation.

```
<xs:element name="RequestProviderInfoUpdateResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="RequestProviderInfoUpdateResult" nillable="true"
xmlns:q5="http://www.microsoft.com/2010/06/Microsoft.WindowsServerSolutions.Common.ProviderFr
amework" type="q5:ArrayOfProviderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.5 IProviderRegistry.GetVersion

This operation returns the provider version number.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="GetVersion">
  <wsdl:input wsam:Action="http://tempuri.org/IProviderRegistry/GetVersion"
message="tns:IProviderRegistry_GetVersion_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IProviderRegistry/GetVersionResponse"
message="tns:IProviderRegistry_GetVersion_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IProviderRegistry_GetVersion_InputMessage** request message, the protocol server returns the version of the protocol server. Upon successful completion of the action, the server MUST respond with **IProviderRegistry_GetVersion_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.5.1 Messages

### 3.6.5.5.1.1 IProviderRegistry_GetVersion_InputMessage

This is the request for the **GetVersion** operation.

```
<wsdl:message name="IProviderRegistry_GetVersion_InputMessage">
  <wsdl:part name="parameters" element="tns:GetVersion" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://temporg.uri/IProviderRegistry/GetVersion
```

The body of the SOAP message MUST contain the **GetVersion** element.

### 3.6.5.5.1.2    IProviderRegistry_GetVersion_OutputMessage

This is the response for the **GetVersion** operation.

```
<wsdl:message name="IProviderRegistry_GetVersion_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetVersionResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/GetVersionResponse
```

The body of the SOAP message MUST contain the **GetVersionResponse** element.

### 3.6.5.5.2 Elements

### 3.6.5.5.2.1    GetVersion

This element specifies input values for the **GetVersion** operation.

```
<xs:element name="GetVersion">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.6.5.5.2.2    GetVersionResponse

This element specifies output values for the **GetVersion** operation.

```
<xs:element name="GetVersionResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetVersionResult" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.6  IProviderRegistry.InformOfProviderUpdate

This operation sends an updated **ProviderInfo** object to a connected client. This operation MUST NOT be invoked by the management client and MUST be ignored by the server.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false"
name="InformOfProviderUpdate">
  <wsdl:output wsam:Action="http://tempuri.org/IProviderRegistry/InformOfProviderUpdate"
message="tns:IProviderRegistry_InformOfProviderUpdate_OutputCallbackMessage" />
```

```
            </wsdl:operation>
```

Upon receiving the **IProviderRegistry_InformOfProviderUpdate_OutputCallbackMessage**
request message, the protocol client receives an updated **ProviderInfo** object. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.6.1 Messages

#### 3.6.5.6.1.1    IProviderRegistry_InformOfProviderUpdate_OutputCallbackMessage

This is the request for the **InformOfProviderUpdate** operation.

```
<wsdl:message name="IProviderRegistry_InformOfProviderUpdate_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:InformOfProviderUpdate" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/InformOfProviderUpdate
```

The body of the SOAP message MUST contain the **InformOfProviderUpdate** element.

### 3.6.5.6.2 Elements

#### 3.6.5.6.2.1    InformOfProviderUpdate

This element specifies input values for the **InformOfProviderUpdate** operation.

```
<xs:element name="InformOfProviderUpdate">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="info" nillable="true"
xmlns:q6="http://www.microsoft.com/2010/06/Microsoft.WindowsServerSolutions.Common.ProviderFr
amework" type="q6:ProviderInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.7 INotificationProviderServiceAdminOnly.Connect

This operation connects to the notification provider.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="Connect">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/Connect"
message="tns:INotificationProviderServiceAdminOnly_Connect_InputMessage" />
</wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_Connect_InputMessage** request
message, the protocol server establishes a connection with the protocol client. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.7.1 Messages

### 3.6.5.7.1.1 INotificationProviderServiceAdminOnly_Connect_InputMessage

This is the request for the **Connect** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_Connect_InputMessage">
  <wsdl:part name="parameters" element="tns:Connect" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/Connect
```

The body of the SOAP message MUST contain the **Connect** element.

### 3.6.5.7.2 Elements

### 3.6.5.7.2.1 Connect

This element specifies input values for the **Connect** operation.

```
<xs:element name="Connect">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="name" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="categoryFilter" nillable="true"
xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q1:ArrayOfstring"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.8 INotificationProviderServiceAdminOnly.RaiseNotifications

This operation raises persistent notifications on the provider.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false" name="RaiseNotifications">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/RaiseNotifications"
message="tns:INotificationProviderServiceAdminOnly_RaiseNotifications_InputMessage" />
</wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_RaiseNotifications_InputMessage**
request message, the protocol server raises persistent notifications. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

If any notification already exists, it will be replaced.

### 3.6.5.8.1 Messages

### 3.6.5.8.1.1 INotificationProviderServiceAdminOnly_RaiseNotifications_InputMessage

This is the request for the **RaiseNotifications** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_RaiseNotifications_InputMessage">
  <wsdl:part name="parameters" element="tns:RaiseNotifications" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/RaiseNotifications
```

The body of the SOAP message MUST contain the **RaiseNotifications** element.

### 3.6.5.8.2 Elements

### 3.6.5.8.2.1    RaiseNotifications

This element specifies input values for the **RaiseNotifications** operation.

```
<xs:element name="RaiseNotifications">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="newNotificationListChunk" nillable="true"
type="xs:base64Binary" />
      <xs:element minOccurs="0" name="moreData" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.9  INotificationProviderServiceAdminOnly.ClearNotifications

This operation clears persistent notifications on the provider.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false" name="ClearNotifications">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/ClearNotifications"
message="tns:INotificationProviderServiceAdminOnly_ClearNotifications_InputMessage" />
</wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_ClearNotifications_InputMessage**
request message, the protocol server clears persistent notifications. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.9.1 Messages

### 3.6.5.9.1.1    INotificationProviderServiceAdminOnly_ClearNotifications_InputMessag
e

This is the request for the **ClearNotifications** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_ClearNotifications_InputMessage">
  <wsdl:part name="parameters" element="tns:ClearNotifications" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/ClearNotifications
```

The body of the SOAP message MUST contain the **ClearNotifications** element.

### 3.6.5.9.2 Elements

#### 3.6.5.9.2.1 ClearNotifications

This element specifies input values for the **ClearNotifications** operation.

```
<xs:element name="ClearNotifications">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="oldNotificationListChunk" nillable="true"
type="xs:base64Binary" />
      <xs:element minOccurs="0" name="moreData" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.10    INotificationProviderServiceAdminOnly.RequestNotificationChange

This operation requests a change to the specified persistent notification.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false"
name="RequestNotificationChange">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/RequestNotificationChange"
message="tns:INotificationProviderServiceAdminOnly_RequestNotificationChange_InputMessage" />
</wsdl:operation>
```

Upon receiving the
**INotificationProviderServiceAdminOnly_RequestNotificationChange_InputMessage** request
message, the protocol server changes the persistent notification. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.6.5.10.1    Messages

#### 3.6.5.10.1.1 INotificationProviderServiceAdminOnly_RequestNotificationChange_Inp
utMessage

This is the request for the **RequestNotificationChange** operation.

```
<wsdl:message
name="INotificationProviderServiceAdminOnly_RequestNotificationChange_InputMessage">
  <wsdl:part name="parameters" element="tns:RequestNotificationChange" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/RequestNotificationChange
```

The body of the SOAP message MUST contain the **RequestNotificationChange** element.

### 3.6.5.10.2    Elements

#### 3.6.5.10.2.1  RequestNotificationChange

This element specifies input values for the **RequestNotificationChange** operation.

```
<xs:element name="RequestNotificationChange">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="changeRequestChunk" nillable="true"
type="xs:base64Binary" />
      <xs:element minOccurs="0" name="moreData" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.11      INotificationProviderServiceAdminOnly.FireEvent

This operation sends a one-time notification to be raised on the provider.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false" name="FireEvent">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/FireEvent"
message="tns:INotificationProviderServiceAdminOnly_FireEvent_InputMessage" />
</wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_FireEvent_InputMessage** request message, the protocol server raises a one-time notification. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.6.5.11.1    Messages

#### 3.6.5.11.1.1  INotificationProviderServiceAdminOnly_FireEvent_InputMessage

This is the request for the **FireEvent** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_FireEvent_InputMessage">
  <wsdl:part name="parameters" element="tns:FireEvent" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/FireEvent
```

The body of the SOAP message MUST contain the **FireEvent** element.

#### 3.6.5.11.2    Elements

#### 3.6.5.11.2.1  FireEvent

This element specifies input values for the **FireEvent** operation.

```
<xs:element name="FireEvent">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="oneShotChunk" nillable="true" type="xs:base64Binary" />
      <xs:element minOccurs="0" name="moreData" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.12 INotificationProviderServiceAdminOnly.AddCategoryFilters

This operation indicates that the caller is interested in additional categories of filters.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false" name="AddCategoryFilters">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/AddCategoryFilters"
  message="tns:INotificationProviderServiceAdminOnly_AddCategoryFilters_InputMessage" />
</wsdl:operation>
```

Upon receiving the
**INotificationProviderServiceAdminOnly_AddCategoryFilters_InputMessage** request message,
the protocol server treats the caller as being interested in additional categories of notifications. In the
event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section
2.2.2.1.

#### 3.6.5.12.1 Messages

#### 3.6.5.12.1.1 INotificationProviderServiceAdminOnly_AddCategoryFilters_InputMessage

This is the request for the **AddCategoryFilters** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_AddCategoryFilters_InputMessage">
  <wsdl:part name="parameters" element="tns:AddCategoryFilters" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/AddCategoryFilters
```

The body of the SOAP message MUST contain the **AddCategoryFilters** element.

#### 3.6.5.12.2 Elements

#### 3.6.5.12.2.1 AddCategoryFilters

This element specifies input values for the **AddCategoryFilters** operation.

```
<xs:element name="AddCategoryFilters">
  <xs:complexType>
    <xs:sequence>
```

```
        <xs:element minOccurs="0" name="addFilters" nillable="true"
xmlns:q2="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q2:ArrayOfstring"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.13    INotificationProviderServiceAdminOnly.RemoveCategoryFilters

This operation indicates that the caller is no longer interested in the specified categories of Notifications.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="false"
name="RemoveCategoryFilters">
   <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/RemoveCategoryFilters"
message="tns:INotificationProviderServiceAdminOnly_RemoveCategoryFilters_InputMessage"
/></wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_RemoveCategoryFilters_InputMessage** request message, the protocol server treats the caller as no longer being interested in additional categories of notifications. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.6.5.13.1    Messages

##### 3.6.5.13.1.1 INotificationProviderServiceAdminOnly_RemoveCategoryFilters_InputMessage

This is the request for the **RemoveCategoryFilters** operation.

```
<wsdl:message
name="INotificationProviderServiceAdminOnly_RemoveCategoryFilters_InputMessage">
  <wsdl:part name="parameters" element="tns:RemoveCategoryFilters" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/RemoveCategoryFilters
```

The body of the SOAP message MUST contain the **RemoveCategoryFilters** element.

#### 3.6.5.13.2    Elements

##### 3.6.5.13.2.1 RemoveCategoryFilters

This element specifies input values for the **RemoveCategoryFilters** operation.

```
<xs:element name="RemoveCategoryFilters">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="removeFilters" nillable="true"
xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/Arrays" type="q3:ArrayOfstring"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.6.5.14    INotificationProviderServiceAdminOnly.Disconnect

This operation disconnects from the notification provider.

```
<wsdl:operation msc:isInitiating="false" msc:isTerminating="true" name="Disconnect">
  <wsdl:input
wsam:Action="http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/
3/2010/INotificationProviderService/Disconnect"
message="tns:INotificationProviderServiceAdminOnly_Disconnect_InputMessage" />
</wsdl:operation>
```

Upon receiving the **INotificationProviderServiceAdminOnly_Disconnect_InputMessage** request message, the protocol server disconnects the caller. In the event of a failure, the appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.6.5.14.1    Messages

#### 3.6.5.14.1.1 INotificationProviderServiceAdminOnly_Disconnect_InputMessage

This is the request for the **Disconnect** operation.

```
<wsdl:message name="INotificationProviderServiceAdminOnly_Disconnect_InputMessage">
  <wsdl:part name="parameters" element="tns:Disconnect" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://www.microsoft.com/WindowsServerSolutions/ProviderFramework/Notifications/3/2010/INotif
icationProviderService/Disconnect
```

The body of the SOAP message MUST contain the **Disconnect** element.

#### 3.6.5.14.2    Elements

#### 3.6.5.14.2.1 Disconnect

This element specifies input values for the **Disconnect** operation.

```
<xs:element name="Disconnect">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.6.6  Timer Events

None.

### 3.6.7 Other Local Events

None.

## 3.7 Addin Management Service Contract Details

None.

### 3.7.1 Abstract Data Model

None.

### 3.7.2 Timers

None.

### 3.7.3 Initialization

None.

### 3.7.4 Higher-Layer Triggered Events

None.

### 3.7.5 Message Processing Events and Sequencing Rules

The following table specifies the message processing events and sequencing rules that are associated with this specification.

| OperationContract | Description |
| --- | --- |
| IAddInMetadata.GetPackagePath | Gets the server path where the add-in package will be copied. |
| IAddInMetadata.GetAddInIds | Gets the installed and uninstalled add-ins. |
| IAddInMetadata.GetAddInInformation | Gets details about the add-in. |
| IAddInMetadata.RegisterForChanges | Registers for changes to the add-in infrastructure. |
| IAddInMetadata.OperationInProgress | Checks whether there are any current installs or uninstalls that would block another operation. |
| IAddInMetadata.ChangeOccured | Callback triggered when the metadata for an add-in has changed. |
| IAddInMetadataInstall.CheckCredentials | Checks whether administrator credentials are valid by validating that the user is an administrator. |
| IAddInMetadataInstall.InstallAddIn | Installs the given add-in. |
| IAddInMetadataInstall.DownloadInstallAddIn | Downloads and installs the given add-in. |
| IAddInMetadataInstall.UninstallAddIn | Uninstalls the add-in with the given ID. There is no effect if there is no add-in with that ID. |
| IAddInMetadataInstall.CancelOperation | Cancels the current operation. |

| OperationContract | Description |
|---|---|
| IAddInMetadataInstall.StartUploadAddinFile | Starts the upload of the add-in with the given ID. |
| IAddInMetadataInstall.UploadAddinFile | Uploads a chunk of a file. |
| IAddInMetadataInstall.EndUploadAddinFile | Ends the upload of a file. |
| IAddInMetadataInstall.ProgressChanged | Callback triggered when the progress for the given installation has changed, or the installation is complete. |
| IAddInMetadataInstall.ErrorOccured | Callback triggered when an error has occurred during installation. |
| IAddInMetadataInstall.RebootRequired | Callback triggered if the installation requires a reboot. |
| IAddInMetadataInstall.RestartDashboardRequired | Callback triggered if the installation requires a restart of the dashboard. |

### 3.7.5.1 IAddInMetadata.GetPackagePath

This operation returns the server path where the add-in package will be copied.

```
<wsdl:operation name="GetPackagePath">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadata/GetPackagePath"
message="tns:IAddInMetadata_GetPackagePath_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/GetPackagePathResponse"
message="tns:IAddInMetadata_GetPackagePath_OutputMessage" />
  <wsdl:fault wsam:Action="http://tempuri.org/IAddInMetadata/GetPackagePathAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadata_GetPackagePath_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadata_GetPackagePath_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadata_GetPackagePath_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.7.5.1.1 Messages

##### 3.7.5.1.1.1    IAddInMetadata_GetPackagePath_InputMessage

This is the request for the **GetPackagePath** operation.

```
<wsdl:message name="IAddInMetadata_GetPackagePath_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPackagePath" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetPackagePath
```

The body of the SOAP message MUST contain the **GetPackagePath** element.

##### 3.7.5.1.1.2    IAddInMetadata_GetPackagePath_OutputMessage

This is the response for the **GetPackagePath** operation.

```
<wsdl:message name="IAddInMetadata_GetPackagePath_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPackagePathResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetPackagePathResponse
```

The body of the SOAP message MUST contain the **GetPackagePathResponse** element.

### 3.7.5.1.2 Elements

#### 3.7.5.1.2.1 GetPackagePath

This element specifies input values for the **GetPackagePath** operation.

```
<xs:element name="GetPackagePath">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="packageId"
xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/" type="q1:guid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.7.5.1.2.2 GetPackagePathResponse

This element specifies output values for the **GetPackagePath** operation.

```
<xs:element name="GetPackagePathResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetPackagePathResult" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.2 IAddInMetadata.GetAddInIds

This operation returns the installed and uninstalled add-ins.

```
<wsdl:operation name="GetAddInIds">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInIds"
message="tns:IAddInMetadata_GetAddInIds_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInIdsResponse"
message="tns:IAddInMetadata_GetAddInIds_OutputMessage" />
  <wsdl:fault wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInIdsAddInErrorFault"
name="AddInErrorFault" message="tns:IAddInMetadata_GetAddInIds_AddInErrorFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IAddInMetadata_GetAddInIds_InputMessage** request message, the protocol
server is notified that the connection is terminating. Upon successful completion of the action, the
server MUST respond with **IAddInMetadata_GetAddInIds_OutputMessage**. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.2.1 Messages

### 3.7.5.2.1.1   IAddInMetadata_GetAddInIds_InputMessage

This is the request for the **GetAddInIds** operation.

```
<wsdl:message name="IAddInMetadata_GetAddInIds_InputMessage">
  <wsdl:part name="parameters" element="tns:GetAddInIds" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetAddInIds
```

The body of the SOAP message MUST contain the **GetAddInIds** element.

### 3.7.5.2.1.2   IAddInMetadata_GetAddInIds_OutputMessage

This is the response for the **GetAddInIds** operation.

```
<wsdl:message name="IAddInMetadata_GetAddInIds_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetAddInIdsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetAddInIdsResponse
```

The body of the SOAP message MUST contain the **GetAddInIdsResponse** element.

### 3.7.5.2.2 Elements

### 3.7.5.2.2.1   GetAddInIds

This element specifies input values for the **GetAddInIds** operation.

```
<xs:element name="GetAddInIds">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.2.2.2   GetAddInIdsResponse

This element specifies output values for the **GetAddInIds** operation.

```
<xs:element name="GetAddInIdsResponse">
  <xs:complexType>
   <xs:sequence>
      <xs:element minOccurs="0" name="GetAddInIdsResult" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/System.Collections.ObjectModel"
type="q2:ReadOnlyCollectionOfguid" />
   </xs:sequence>
  </xs:complexType>
```

```
        </xs:element>
```

### 3.7.5.3 IAddInMetadata.GetAddInInformation

This operation returns details about an add-in.

```
<wsdl:operation name="GetAddInInformation">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInInformation"
message="tns:IAddInMetadata_GetAddInInformation_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInInformationResponse"
message="tns:IAddInMetadata_GetAddInInformation_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadata/GetAddInInformationAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadata_GetAddInInformation_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadata_GetAddInInformation_InputMessage** request message, the
protocol server is notified that the connection is terminating. Upon successful completion of the action,
the server MUST respond with **IAddInMetadata_GetAddInInformation_OutputMessage**. In the
event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.7.5.3.1 Messages

##### 3.7.5.3.1.1 IAddInMetadata_GetAddInInformation_InputMessage

This is the request for the **GetAddInInformation** operation.

```
<wsdl:message name="IAddInMetadata_GetAddInInformation_InputMessage">
  <wsdl:part name="parameters" element="tns:GetAddInInformation" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetAddInInformation
```

The body of the SOAP message MUST contain the **GetAddInInformation** element.

##### 3.7.5.3.1.2 IAddInMetadata_GetAddInInformation_OutputMessage

This is the response for the **GetAddInInformation** operation.

```
<wsdl:message name="IAddInMetadata_GetAddInInformation_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetAddInInformationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/GetAddInInformationResponse
```

The body of the SOAP message MUST contain the **GetAddInInformationResponse** element.

#### 3.7.5.3.2 Elements

### 3.7.5.3.2.1    GetAddInInformation

This element specifies input values for the **GetAddInInformation** operation.

```
<xs:element name="GetAddInInformation">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="packageId"
xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/" type="q3:guid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.3.2.2    GetAddInInformationResponse

This element specifies output values for the **GetAddInInformation** operation.

```
<xs:element name="GetAddInInformationResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetAddInInformationResult" nillable="true"
xmlns:q4="http://schemas.microsoft.com/WindowsServerSolutions/2010/03/Addins"
type="q4:PackageInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.7.5.4  IAddInMetadata.RegisterForChanges

This operation registers for changes to an add-in.

```
<wsdl:operation name="RegisterForChanges">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadata/RegisterForChanges"
message="tns:IAddInMetadata_RegisterForChanges_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/RegisterForChangesResponse"
message="tns:IAddInMetadata_RegisterForChanges_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadata_RegisterForChanges_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadata_RegisterForChanges_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.4.1 Messages

### 3.7.5.4.1.1    IAddInMetadata_RegisterForChanges_InputMessage

This is the request for the **RegisterForChanges** operation.

```
<wsdl:message name="IAddInMetadata_RegisterForChanges_InputMessage">
  <wsdl:part name="parameters" element="tns:RegisterForChanges" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/RegisterForChanges
```

The body of the SOAP message MUST contain the **RegisterForChanges** element.

### 3.7.5.4.1.2    IAddInMetadata_RegisterForChanges_OutputMessage

This is the response for the **RegisterForChanges** operation.

```
<wsdl:message name="IAddInMetadata_RegisterForChanges_OutputMessage">
  <wsdl:part name="parameters" element="tns:RegisterForChangesResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/RegisterForChangesResponse
```

The body of the SOAP message MUST contain the **RegisterForChangesResponse** element.

### 3.7.5.4.2 Elements

### 3.7.5.4.2.1    RegisterForChanges

This element specifies input values for the **RegisterForChanges** operation.

```
<xs:element name="RegisterForChanges">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="server" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.4.2.2    RegisterForChangesResponse

This element specifies output values for the **RegisterForChanges** operation.

```
<xs:element name="RegisterForChangesResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.5  IAddInMetadata.OperationInProgress

This operation checks whether there are any active install or uninstall processes that would block another operation.

```
<wsdl:operation name="OperationInProgress">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadata/OperationInProgress"
message="tns:IAddInMetadata_OperationInProgress_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/OperationInProgressResponse"
message="tns:IAddInMetadata_OperationInProgress_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadata/OperationInProgressAddInErrorFault"
```

```
name="AddInErrorFault"
message="tns:IAddInMetadata_OperationInProgress_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadata_OperationInProgress_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadata_OperationInProgress_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.5.1 Messages

#### 3.7.5.5.1.1    IAddInMetadata_OperationInProgress_InputMessage

This is the request for the **OperationInProgress** operation.

```
<wsdl:message name="IAddInMetadata_OperationInProgress_InputMessage">
  <wsdl:part name="parameters" element="tns:OperationInProgress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/OperationInProgress
```

The body of the SOAP message MUST contain the **OperationInProgress** element.

#### 3.7.5.5.1.2    IAddInMetadata_OperationInProgress_OutputMessage

This is the response for the **OperationInProgress** operation.

```
<wsdl:message name="IAddInMetadata_OperationInProgress_OutputMessage">
  <wsdl:part name="parameters" element="tns:OperationInProgressResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/OperationInProgressResponse
```

The body of the SOAP message MUST contain the **OperationInProgressResponse** element.

### 3.7.5.5.2 Elements

#### 3.7.5.5.2.1    OperationInProgress

This element specifies input values for the **OperationInProgress** operation.

```
<xs:element name="OperationInProgress">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.7.5.5.2.2    OperationInProgressResponse

This element specifies output values for the **OperationInProgress** operation.

```
<xs:element name="OperationInProgressResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="OperationInProgressResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.6 IAddInMetadata.ChangeOccured

This callback operation is called when a change has occurred.

```
<wsdl:operation name="ChangeOccured">
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadata/ChangeOccured"
message="tns:IAddInMetadata_ChangeOccured_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.7.5.6.1 Messages

### 3.7.5.6.1.1    IAddInMetadata_ChangeOccured_OutputCallbackMessage

This is the request for the **ChangeOccured** operation.

```
<wsdl:message name="IAddInMetadata_ChangeOccured_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ChangeOccured" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadata/ChangeOccured
```

The body of the SOAP message MUST contain the **ChangeOccured** element.

### 3.7.5.6.2 Elements

### 3.7.5.6.2.1    ChangeOccured

 This element specifies input values for the **ChangeOccured** operation.

```
<xs:element name="ChangeOccured">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.7 IAddInMetadataInstall.CheckCredentials

This operation validates that the user is an administrator.

```
<wsdl:operation name="CheckCredentials">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/CheckCredentials"
message="tns:IAddInMetadataInstall_CheckCredentials_InputMessage" />
```

```
      <wsdl:output
wsam:Action="http://tempuri.org/IAddInMetadataInstall/CheckCredentialsResponse"
message="tns:IAddInMetadataInstall_CheckCredentials_OutputMessage" />
      <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadataInstall/CheckCredentialsAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadataInstall_CheckCredentials_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_CheckCredentials_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadataInstall_CheckCredentials_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.7.1 Messages

#### 3.7.5.7.1.1    IAddInMetadataInstall_CheckCredentials_InputMessage

This is the request for the **CheckCredentials** operation.

```
<wsdl:message name="IAddInMetadataInstall_CheckCredentials_InputMessage">
  <wsdl:part name="parameters" element="tns:CheckCredentials" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/CheckCredentials
```

The body of the SOAP message MUST contain the **CheckCredentials** element.

#### 3.7.5.7.1.2    IAddInMetadataInstall_CheckCredentials_OutputMessage

This is the response for the **CheckCredentials** operation.

```
<wsdl:message name="IAddInMetadataInstall_CheckCredentials_OutputMessage">
  <wsdl:part name="parameters" element="tns:CheckCredentialsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/CheckCredentialsResponse
```

The body of the SOAP message MUST contain the **CheckCredentialsResponse** element.

### 3.7.5.7.2 Elements

#### 3.7.5.7.2.1    CheckCredentials

This element specifies input values for the **CheckCredentials** operation.

```
<xs:element name="CheckCredentials">
  <xs:complexType>
    <xs:sequence />
```

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

### 3.7.5.7.2.2    CheckCredentialsResponse

This element specifies output values for the **CheckCredentials** operation.

```
<xs:element name="CheckCredentialsResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.7.5.8  IAddInMetadataInstall.InstallAddIn

This operation installs the given add-in.

```
<wsdl:operation name="InstallAddIn">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/InstallAddIn"
message="tns:IAddInMetadataInstall_InstallAddIn_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_InstallAddIn_InputMessage** request message, the protocol server is notified that the connection is terminating. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.8.1 Messages

#### 3.7.5.8.1.1    IAddInMetadataInstall_InstallAddIn_InputMessage

This is the request for the **InstallAddIn** operation.

```
<wsdl:message name="IAddInMetadataInstall_InstallAddIn_InputMessage">
  <wsdl:part name="parameters" element="tns:InstallAddIn" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/InstallAddIn
```

The body of the SOAP message MUST contain the **InstallAddIn** element.

### 3.7.5.8.2 Elements

#### 3.7.5.8.2.1    InstallAddIn

This element specifies input values for the **InstallAddIn** operation.

```
<xs:element name="InstallAddIn">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="addin" nillable="true"
xmlns:q1="http://schemas.microsoft.com/WindowsServerSolutions/2010/03/Addins"
type="q1:Package" />
```

```
        <xs:element minOccurs="0" name="installOnClients" nillable="true"
xmlns:q2="http://schemas.microsoft.com/WindowsServerSolutions/2010/03/Addins"
type="q2:ClientInstallFilter" />
      </xs:sequence>
    </xs:complexType>
</xs:element>
```

### 3.7.5.9  IAddInMetadataInstall.DownloadInstallAddIn

This operation downloads and installs the given add-in.

```
<wsdl:operation name="DownloadInstallAddIn">
    <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/DownloadInstallAddIn"
message="tns:IAddInMetadataInstall_DownloadInstallAddIn_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_DownloadInstallAddIn_InputMessage** request message, the protocol server is notified that the connection is terminating. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.9.1 Messages

#### 3.7.5.9.1.1   IAddInMetadataInstall_DownloadInstallAddIn_InputMessage

This is the request for the **DownloadInstallAddIn** operation.

```
<wsdl:message name="IAddInMetadataInstall_DownloadInstallAddIn_InputMessage">
    <wsdl:part name="parameters" element="tns:DownloadInstallAddIn" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/DownloadInstallAddIn
```

The body of the SOAP message MUST contain the **DownloadInstallAddIn** element.

### 3.7.5.9.2 Elements

#### 3.7.5.9.2.1   DownloadInstallAddIn

This element specifies input values for the **DownloadInstallAddIn** operation.

```
<xs:element name="DownloadInstallAddIn">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="address" nillable="true" type="xs:anyURI" />
      <xs:element minOccurs="0" name="certThumbprint" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="expectedPackageId"
xmlns:q3="http://schemas.microsoft.com/2003/10/Serialization/" type="q3:guid" />
      <xs:element minOccurs="0" name="installOnClients" nillable="true"
xmlns:q4="http://schemas.microsoft.com/WindowsServerSolutions/2010/03/Addins"
type="q4:ClientInstallFilter" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.10    IAddInMetadataInstall.UninstallAddIn

This operation uninstalls the given add-in. This operation has no effect if there is no add-in that matches the supplied ID.

```
<wsdl:operation name="UninstallAddIn">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/UninstallAddIn"
message="tns:IAddInMetadataInstall_UninstallAddIn_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_UninstallAddIn_InputMessage** request message, the protocol server is notified that the connection is terminating. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.7.5.10.1    Messages

#### 3.7.5.10.1.1 IAddInMetadataInstall_UninstallAddIn_InputMessage

This is the request for the **UninstallAddIn** operation.

```
<wsdl:message name="IAddInMetadataInstall_UninstallAddIn_InputMessage">
  <wsdl:part name="parameters" element="tns:UninstallAddIn" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/UninstallAddIn
```

The body of the SOAP message MUST contain the **UninstallAddIn** element.

#### 3.7.5.10.2    Elements

#### 3.7.5.10.2.1 UninstallAddIn

This element specifies input values for the **UninstallAddIn** operation.

```
<xs:element name="UninstallAddIn">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="id"
xmlns:q5="http://schemas.microsoft.com/2003/10/Serialization/" type="q5:guid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.11    IAddInMetadataInstall.CancelOperation

This operation cancels the current operation.

```
<wsdl:operation name="CancelOperation">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/CancelOperation"
message="tns:IAddInMetadataInstall_CancelOperation_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_CancelOperation_InputMessage** request message, the protocol server is notified that the connection is terminating. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.11.1    Messages

### 3.7.5.11.1.1 IAddInMetadataInstall_CancelOperation_InputMessage

This is the request for the **CancelOperation** operation.

```
<wsdl:message name="IAddInMetadataInstall_CancelOperation_InputMessage">
  <wsdl:part name="parameters" element="tns:CancelOperation" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/CancelOperation
```

The body of the SOAP message MUST contain the **CancelOperation** element.

### 3.7.5.11.2    Elements

### 3.7.5.11.2.1 CancelOperation

This element specifies input values for the **CancelOperation** operation.

```
<xs:element name="CancelOperation">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.12      IAddInMetadataInstall.StartUploadAddinFile

This operation starts the upload of the add-in with the given ID value.

```
<wsdl:operation name="StartUploadAddinFile">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/StartUploadAddinFile"
message="tns:IAddInMetadataInstall_StartUploadAddinFile_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAddInMetadataInstall/StartUploadAddinFileResponse"
message="tns:IAddInMetadataInstall_StartUploadAddinFile_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadataInstall/StartUploadAddinFileAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadataInstall_StartUploadAddinFile_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_StartUploadAddinFile_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadataInstall_StartUploadAddinFile_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.12.1    Messages

### 3.7.5.12.1.1 IAddInMetadataInstall_StartUploadAddinFile_InputMessage

This is the request for the **StartUploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_StartUploadAddinFile_InputMessage">
  <wsdl:part name="parameters" element="tns:StartUploadAddinFile" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/StartUploadAddinFile
```

The body of the SOAP message MUST contain the **StartUploadAddinFile** element.

### 3.7.5.12.1.2 IAddInMetadataInstall_StartUploadAddinFile_OutputMessage

This is the response for the **StartUploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_StartUploadAddinFile_OutputMessage">
  <wsdl:part name="parameters" element="tns:StartUploadAddinFileResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/StartUploadAddinFileResponse
```

The body of the SOAP message MUST contain the **StartUploadAddinFileResponse** element.

### 3.7.5.12.2    Elements

### 3.7.5.12.2.1 StartUploadAddinFile

This element specifies input values for the **StartUploadAddinFile** operation.

```
<xs:element name="StartUploadAddinFile">
  <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" name="addinId"
xmlns:q6="http://schemas.microsoft.com/2003/10/Serialization/" type="q6:guid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.12.2.2 StartUploadAddinFileResponse

This element specifies output values for the **StartUploadAddinFile** operation.

```
<xs:element name="StartUploadAddinFileResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="StartUploadAddinFileResult"
xmlns:q7="http://schemas.microsoft.com/2003/10/Serialization/" type="q7:guid" />
    </xs:sequence>
  </xs:complexType>
```

```
      </xs:element>
```

### 3.7.5.13    IAddInMetadataInstall.UploadAddinFile

This operation uploads a chunk of an add-in file.

```
<wsdl:operation name="UploadAddinFile">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/UploadAddinFile"
message="tns:IAddInMetadataInstall_UploadAddinFile_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadataInstall/UploadAddinFileResponse"
message="tns:IAddInMetadataInstall_UploadAddinFile_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadataInstall/UploadAddinFileAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadataInstall_UploadAddinFile_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_UploadAddinFile_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **IAddInMetadataInstall_UploadAddinFile_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.7.5.13.1    Messages

#### 3.7.5.13.1.1  IAddInMetadataInstall_UploadAddinFile_InputMessage

This is the request for the **UploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_UploadAddinFile_InputMessage">
  <wsdl:part name="parameters" element="tns:UploadAddinFile" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/UploadAddinFile
```

The body of the SOAP message MUST contain the **UploadAddinFile** element.

#### 3.7.5.13.1.2  IAddInMetadataInstall_UploadAddinFile_OutputMessage

This is the response for the **UploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_UploadAddinFile_OutputMessage">
  <wsdl:part name="parameters" element="tns:UploadAddinFileResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/UploadAddinFileResponse
```

The body of the SOAP message MUST contain the **UploadAddinFileResponse** element.

### 3.7.5.13.2    Elements

### 3.7.5.13.2.1  UploadAddinFile

This element specifies input values for the **UploadAddinFile** operation.

```
<xs:element name="UploadAddinFile">
  <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" name="sessionId"
xmlns:q8="http://schemas.microsoft.com/2003/10/Serialization/" type="q8:guid" />
      <xs:element minOccurs="0" name="data" nillable="true" type="xs:base64Binary" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.13.2.2  UploadAddinFileResponse

This element specifies output values for the **UploadAddinFile** operation.

```
<xs:element name="UploadAddinFileResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.14       IAddInMetadataInstall.EndUploadAddinFile

This operation ends the upload of an add-in file.

```
<wsdl:operation name="EndUploadAddinFile">
  <wsdl:input wsam:Action="http://tempuri.org/IAddInMetadataInstall/EndUploadAddinFile"
message="tns:IAddInMetadataInstall_EndUploadAddinFile_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IAddInMetadataInstall/EndUploadAddinFileResponse"
message="tns:IAddInMetadataInstall_EndUploadAddinFile_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IAddInMetadataInstall/EndUploadAddinFileAddInErrorFault"
name="AddInErrorFault"
message="tns:IAddInMetadataInstall_EndUploadAddinFile_AddInErrorFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IAddInMetadataInstall_EndUploadAddinFile_InputMessage** request
message, the protocol server is notified that the connection is terminating. Upon successful completion
of the action, the server MUST respond with
**IAddInMetadataInstall_EndUploadAddinFile_OutputMessage**. In the event of a failure, an
appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.7.5.14.1    Messages

### 3.7.5.14.1.1  IAddInMetadataInstall_EndUploadAddinFile_InputMessage

This is the request for the **EndUploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_EndUploadAddinFile_InputMessage">
  <wsdl:part name="parameters" element="tns:EndUploadAddinFile" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/EndUploadAddinFile
```

The body of the SOAP message MUST contain the **EndUploadAddinFile** element.

### 3.7.5.14.1.2 IAddInMetadataInstall_EndUploadAddinFile_OutputMessage

This is the response for the **EndUploadAddinFile** operation.

```
<wsdl:message name="IAddInMetadataInstall_EndUploadAddinFile_OutputMessage">
  <wsdl:part name="parameters" element="tns:EndUploadAddinFileResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/EndUploadAddinFileResponse
```

The body of the SOAP message MUST contain the **EndUploadAddinFileResponse** element.

### 3.7.5.14.2    Elements

### 3.7.5.14.2.1 EndUploadAddinFile

This element specifies input values for the EndUploadAddinFile operation.

```
<xs:element name="EndUploadAddinFile">
  <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" name="sessionId"
xmlns:q9="http://schemas.microsoft.com/2003/10/Serialization/" type="q9:guid" />
   </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.14.2.2 EndUploadAddinFileResponse

This element specifies output values for the **EndUploadAddinFile** operation.

```
<xs:element name="EndUploadAddinFileResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.5.15      IAddInMetadataInstall.ProgressChanged

This callback operation is triggered when the progress for the given operation has changed, or the operation has completed.

```
<wsdl:operation name="ProgressChanged">
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadataInstall/ProgressChanged"
message="tns:IAddInMetadataInstall_ProgressChanged_OutputCallbackMessage" />
```

```
</wsdl:operation>
```

### 3.7.5.15.1    Messages

### 3.7.5.15.1.1  IAddInMetadataInstall_ProgressChanged_OutputCallbackMessage

This is the request for the **ProgressChanged** operation.

```
<wsdl:message name="IAddInMetadataInstall_ProgressChanged_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ProgressChanged" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/ProgressChanged
```

The body of the SOAP message MUST contain the **ProgressChanged** element.

### 3.7.5.15.2    Elements

### 3.7.5.15.2.1  ProgressChanged

This element specifies input values for the **ProgressChanged** operation.

```
<xs:element name="ProgressChanged">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="p" type="xs:int" />
      <xs:element minOccurs="0" name="completed" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.16       IAddInMetadataInstall.ErrorOccured

This callback operation is triggered when an error has occurred during the operation.

```
<wsdl:operation name="ErrorOccured">
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadataInstall/ErrorOccured"
message="tns:IAddInMetadataInstall_ErrorOccured_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.7.5.16.1    Messages

### 3.7.5.16.1.1  IAddInMetadataInstall_ErrorOccured_OutputCallbackMessage

This is the request for the **ErrorOccured** operation.

```
<wsdl:message name="IAddInMetadataInstall_ErrorOccured_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:ErrorOccured" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/ErrorOccured
```

The body of the SOAP message MUST contain the **ErrorOccured** element.

### 3.7.5.16.2    Elements

#### 3.7.5.16.2.1  ErrorOccured

This element specifies input values for the **ErrorOccured** operation.

```
<xs:element name="ErrorOccured">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="messageId"
xmlns:q10="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.AddinInfr
astructure" type="q10:AddInError" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.7.5.17        IAddInMetadataInstall.RebootRequired

This callback operation is triggered if the operation required a reboot.

```
<wsdl:operation name="RebootRequired">
  <wsdl:output wsam:Action="http://tempuri.org/IAddInMetadataInstall/RebootRequired"
message="tns:IAddInMetadataInstall_RebootRequired_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.7.5.17.1    Messages

#### 3.7.5.17.1.1  IAddInMetadataInstall_RebootRequired_OutputCallbackMessage

This is the request for the **RebootRequired** operation.

```
<wsdl:message name="IAddInMetadataInstall_RebootRequired_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RebootRequired" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/RebootRequired
```

The body of the SOAP message MUST contain the **RebootRequired** element.

### 3.7.5.17.2    Elements

#### 3.7.5.17.2.1  RebootRequired

This element specifies input values for the **RebootRequired** operation.

```
<xs:element name="RebootRequired">
  <xs:complexType>
```

```
      <xs:sequence />
    </xs:complexType>
  </xs:element>
```

### 3.7.5.18    IAddInMetadataInstall.RestartDashboardRequired

This callback operation is triggered if the operation required a restart of the dashboard.

```
<wsdl:operation name="RestartDashboardRequired">
  <wsdl:output
wsam:Action="http://tempuri.org/IAddInMetadataInstall/RestartDashboardRequired"
message="tns:IAddInMetadataInstall_RestartDashboardRequired_OutputCallbackMessage" />
</wsdl:operation>
```

#### 3.7.5.18.1    Messages

##### 3.7.5.18.1.1  IAddInMetadataInstall_RestartDashboardRequired_OutputCallbackMessage

This is the request for the **RestartDashboardRequired** operation.

```
<wsdl:message name="IAddInMetadataInstall_RestartDashboardRequired_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RestartDashboardRequired" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IAddInMetadataInstall/RestartDashboardRequired
```

The body of the SOAP message MUST contain the **RestartDashboardRequired** element.

#### 3.7.5.18.2    Elements

##### 3.7.5.18.2.1  RestartDashboardRequired

This element specifies input values for the **RestartDashboardRequired** operation.

```
<xs:element name="RestartDashboardRequired">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.7.6  Timer Events

None.

### 3.7.7  Other Local Events

None.

### 3.8 Network Service Contract Details

### 3.8.1 Abstract Data Model

None.

### 3.8.2 Timers

None.

### 3.8.3 Initialization

None.

### 3.8.4 Higher-Layer Triggered Events

None.

### 3.8.5 Message Processing Events and Sequencing Rules

Note: Some of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it in the Product Behavior appendix.

| OperationContract | Description |
|---|---|
| INetworkingService.UpdateClientAddress | Called by the client to notify the server of the client's current IP address. |
| INetworkingService.RemoveClient | Called when the client is permanently removed in the dashboard. Requires administrator access and can only be called locally. |
| INetworkingService.Ping | Used to implement IsServerOnline API. |
| INetworkingService.GetVPNServerSetting | Used to retrieve VPN server settings. |
| INetworkingService.GetVirtualNetworkSettings | Used to retrieve Azure Virtual Network settings. |

### 3.8.5.1 INetworkingService.UpdateClientAddress

This operation notifies the server of the client's current IP address.

```
<wsdl:operation name="UpdateClientAddress">
  <wsdl:input wsam:Action="http://tempuri.org/INetworkingService/UpdateClientAddress"
message="tns:INetworkingService_UpdateClientAddress_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/INetworkingService/UpdateClientAddressResponse"
message="tns:INetworkingService_UpdateClientAddress_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/INetworkingService/UpdateClientAddressArgumentExceptionFault"
name="ArgumentExceptionFault"
message="tns:INetworkingService_UpdateClientAddress_ArgumentExceptionFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **INetworkingService_UpdateClientAddress_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **INetworkingService_UpdateClientAddress_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.8.5.1.1 Messages

#### 3.8.5.1.1.1   INetworkingService_UpdateClientAddress_InputMessage

This is the request for the **UpdateClientAddress** operation.

```
<wsdl:message name="INetworkingService_UpdateClientAddress_InputMessage">
  <wsdl:part name="parameters" element="tns:UpdateClientAddress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/INetworkingService/UpdateClientAddress
```

The body of the SOAP message MUST contain the **UpdateClientAddress** element.

#### 3.8.5.1.1.2   INetworkingService_UpdateClientAddress_OutputMessage

This is the response for the **UpdateClientAddress** operation.

```
<wsdl:message name="IProviderRegistry_UpdateClientAddress_OutputMessage">
  <wsdl:part name="parameters" element="tns:UpdateClientAddressResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/UpdateClientAddressResponse
```

The body of the SOAP message MUST contain the **UpdateClientAddressResponse** element.

### 3.8.5.1.2 Elements

#### 3.8.5.1.2.1   UpdateClientAddress

This element specifies input values for the **UpdateClientAddress** operation.

```
<xs:element name="UpdateClientAddress">
  <xs:complexType>
    <xs:element minOccurs="0" name="client" nillable="true" type="xs:string" />
  </xs:complexType>
</xs:element>
```

#### 3.8.5.1.2.2   UpdateClientAddressResponse

This element specifies output values for the **UpdateClientAddress** operation.

```
<xs:element name="UpdateClientAddressResponse">
```

```
      <xs:complexType>
        <xs:sequence />
      </xs:complexType>
    </xs:element>
```

## 3.8.5.2 INetworkingService.RemoveClient

This operation permanently removes a client from the dashboard. This operation requires administrator access and can only be called locally.

```
    <wsdl:operation name="RemoveClient">
      <wsdl:input wsam:Action="http://tempuri.org/INetworkingService/RemoveClient"
    message="tns:INetworkingService_RemoveClient_InputMessage" />
      <wsdl:output wsam:Action="http://tempuri.org/INetworkingService/RemoveClientResponse"
    message="tns:INetworkingService_RemoveClient_OutputMessage" />
      <wsdl:fault
    wsam:Action="http://tempuri.org/INetworkingService/RemoveClientArgumentExceptionFault"
    name="ArgumentExceptionFault"
    message="tns:INetworkingService_RemoveClient_ArgumentExceptionFault_FaultMessage" />
    </wsdl:operation>
```

Upon receiving the **INetworkingService_RemoveClient_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **INetworkingService_RemoveClient_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.8.5.2.1 Messages

#### 3.8.5.2.1.1 INetworkingService_RemoveClient_InputMessage

This is the request for the **RemoveClient** operation.

```
    <wsdl:message name="INetworkingService_RemoveClient_InputMessage">
      <wsdl:part name="parameters" element="tns:RemoveClient" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/INetworkingService/RemoveClient
```

The body of the SOAP message MUST contain the **RemoveClient** element.

#### 3.8.5.2.1.2 INetworkingService_RemoveClient_OutputMessage

This is the response for the **RemoveClient** operation.

```
    <wsdl:message name="IProviderRegistry_RemoveClient_OutputMessage">
      <wsdl:part name="parameters" element="tns:RemoveClientResponse" />
    </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IProviderRegistry/RemoveClientResponse
```

The body of the SOAP message MUST contain the **RemoveClientResponse** element.

### 3.8.5.2.2 Elements

#### 3.8.5.2.2.1 RemoveClient

This element specifies input values for the **RemoveClient** operation.

```
<xs:element name="RemoveClient">
  <xs:complexType>
    <xs:element minOccurs="0" name="client" nillable="true" type="xs:string" />
  </xs:complexType>
</xs:element>
```

#### 3.8.5.2.2.2 RemoveClientResponse

This element specifies output values for the **RemoveClient** operation.

```
<xs:element name="RemoveClientResponse">
  <xs:complexType>
   <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.8.5.3 INetworkingService.Ping

This operation is used to implement the **IsServerOnline** API.

```
<wsdl:operation name="Ping">
  <wsdl:input wsam:Action="http://tempuri.org/INetworkingService/Ping"
message="tns:INetworkingService_Ping_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/INetworkingService/PingResponse"
message="tns:INetworkingService_Ping_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **INetworkingService_Ping_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **INetworkingService_Ping_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.8.5.3.1 Messages

#### 3.8.5.3.1.1 INetworkingService_Ping_InputMessage

This is the request for the **Ping** operation.

```
<wsdl:message name="INetworkingService_Ping_InputMessage">
  <wsdl:part name="parameters" element="tns:Ping" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/INetworkingService/Ping
```

The body of the SOAP message MUST contain the **Ping** element.

### 3.8.5.3.1.2   INetworkingService_Ping_OutputMessage

This is the response for the **Ping** operation.

```
<wsdl:message name="IProviderRegistry_Ping_OutputMessage">
  <wsdl:part name="parameters" element="tns:PingResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/PingResponse
```

The body of the SOAP message MUST contain the **PingResponse** element.

### 3.8.5.3.2 Elements

### 3.8.5.3.2.1   Ping

This element specifies input values for the **Ping** operation.

```
<xs:element name="Ping">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.8.5.3.2.2   PingResponse

This element specifies output values for the **Ping** operation.

```
<xs:element name="PingResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.8.5.4  INetworkingService.GetVPNServerSetting

This operation returns the server's VPN settings.

```
<wsdl:operation name="GetVPNServerSetting">
  <wsdl:input wsam:Action="http://tempuri.org/INetworkingService/GetVPNServerSetting"
message="tns:INetworkingService_GetVPNServerSetting_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/INetworkingService/GetVPNServerSettingResponse"
message="tns:INetworkingService_GetVPNServerSetting_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **INetworkingService_GetVPNServerSetting_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with

**INetworkingService_GetVPNServerSetting_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.8.5.4.1 Messages

#### 3.8.5.4.1.1    INetworkingService_GetVPNServerSetting_InputMessage

This is the request for the **GetVPNServerSetting** operation.

```
<wsdl:message name="INetworkingService_GetVPNServerSetting_InputMessage">
  <wsdl:part name="parameters" element="tns:GetVPNServerSetting" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/INetworkingService/GetVPNServerSetting
```

The body of the SOAP message MUST contain the **GetVPNServerSetting** element.

#### 3.8.5.4.1.2    INetworkingService_GetVPNServerSetting_OutputMessage

This is the response for the **GetVPNServerSetting** operation.

```
<wsdl:message name="IProviderRegistry_GetVPNServerSetting_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetVPNServerSettingResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/GetVPNServerSettingResponse
```

The body of the SOAP message MUST contain the **GetVPNServerSettingResponse** element.

### 3.8.5.4.2 Elements

#### 3.8.5.4.2.1    GetVPNServerSetting

This element specifies input values for the **GetVPNServerSetting** operation.

```
<xs:element name="GetVPNServerSetting">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.8.5.4.2.2    GetVPNServerSettingResponse

This element specifies output values for the **GetVPNServerSetting** operation.

```
<xs:element name="GetVPNServerSettingResponse">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" name="GetVPNServerSettingResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss" type="q1:VPNSetting" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.8.5.5 INetworkService.GetVirtualNetworkSettings

~~Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

This operation returns the server's Azure Virtual Network settings~~.~~ (section 2.2.4.8.2).

```
<wsdl:operation name="GetVirtualNetworkSettings">
  <wsdl:input wsam:Action="http://tempuri.org/INetworkingService/GetVirtualNetworkSettings"
message="tns:INetworkingService_GetVirtualNetworkSettings_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/INetworkingService/GetVirtualNetworkSettings"
message="tns:INetworkingService_GetVirtualNetworkSettings_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **INetworkingService_GetVirtualNetworkSettings_InputMessage** request message, the protocol server is notified that the connection is terminating. Upon successful completion of the action, the server MUST respond with **INetworkingService_GetVirtualNetworkSettings_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.8.5.5.1 Messages

#### 3.8.5.5.1.1 INetworkService_GetVirtualNetworkSettings_InputMessage

~~Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

This is the request for the **GetVirtualNetworkSettings** operation.

```
<wsdl:message name="INetworkingService_GetVirtualNetworkSettings_InputMessage">
  <wsdl:part name="parameters" element="tns:GetVirtualNetworkSettings" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/INetworkingService/GetVirtualNetworkSettings
```

The body of the SOAP message MUST contain the **GetVirtualNetworkSettings** element.

#### 3.8.5.5.1.2 INetworkService_GetVirtualNetworkSettings_OutputMessage

~~Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

This is the response for the **GetVirtualNetworkSettings** operation.

```
<wsdl:message name="IProviderRegistry_GetVirtualNetworkSettings_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetVirtualNetworkSettingsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProviderRegistry/GetVirtualNetworkSettingsResponse
```

The body of the SOAP message MUST contain the **GetVirtualNetworkSettingsResponse** element.

### 3.8.5.5.2 Elements

#### 3.8.5.5.2.1    GetVirtualNetworkSettings

Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.

This element specifies input values for the **GetVirtualNetworkSettings** operation.

```
<xs:element name="GetVirtualNetworkSettings">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.8.5.5.2.2    GetVirtualNetworkSettingsResponse

Note: All of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.

This element specifies output values for the **GetVirtualNetworkSettings** operation.

```
<xs:element name="GetVirtualNetworkSettingsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name=" GetVirtualNetworkSettingsResult" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss" type="q1:VirtualNetworkSettings" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.8.6  Timer Events

None.

### 3.8.7  Other Local Events

None.

### 3.9   Domain Infrastructure Service Contract Details

### 3.9.1   Abstract Data Model

None.

### 3.9.2   Timers

None.

### 3.9.3   Initialization

None.

### 3.9.4   Higher-Layer Triggered Events

None.

### 3.9.5   Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IDomainMaintenanceManager.SetCredentials | Set and validate user credentials for current domain configuration. This API is intended for repair scenarios when credentials provided previously become stale (for example, a password has changed) and need to be updated before domain maintenance can continue successfully. |
| IDomainMaintenanceManager.ValidateCredentials | Validate the current user credentials. |
| IDomainMaintenanceManager.UpdateDnsRecords | Update DNS records for the domain. |
| IDomainMaintenanceManager.QueryAddressFromDNS | Determine the public IP address (IPv4 or IPv6) corresponding to the currently configured domain name using public DNS infrastructure. |
| IDomainMaintenanceManager.QueryAddressFromProvider | Query the active maintenance provider for the server's public IP address. |
| IDomainMaintenanceManager.TestConnection | Verify whether the connection to the current provider's backend service can be established. This method ~~should be~~is used for diagnostics only, as no guarantees of network/service conditions can be made immediately after this method returns. |
| IDomainMaintenanceManager.GetDomainState | Query the active maintenance provider for current domain state. |
| IDomainMaintenanceManager.GetCertificateState | Quey the active maintenance provider for current certificate state. |
| IDomainMaintenanceManager.GetDomainNameConfiguration | Return the currently active domain name configuration. |
| IDomainMaintenanceManager.GetDynamicDNSInfo | Return data about dynamic DNS updates (the time of the most recent successful update, for example). |

| OperationContract | Description |
|---|---|
| IDomainMaintenanceManager.GetSettings | Return current domain manager settings. |

#### 3.9.5.1 IDomainMaintenanceManager.SetCredentials

This operation sets and validates user credentials for the current domain configuration.

```
<wsdl:operation name="SetCredentials">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/SetCredentials"
message="tns:IDomainMaintenanceManager_SetCredentials_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/SetCredentialsResponse"
message="tns:IDomainMaintenanceManager_SetCredentials_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/SetCredentialsDomainManagerFaultFau
lt" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_SetCredentials_DomainManagerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_SetCredentials_InputMessage** request message, the protocol server sets and validates user credentials. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_SetCredentials_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.9.5.1.1 Messages

##### 3.9.5.1.1.1   IDomainMaintenanceManager_SetCredentials_InputMessage

This is the request for the **SetCredentials** operation.

```
<wsdl:message name="IDomainMaintenanceManager_SetCredentials_InputMessage">
  <wsdl:part name="parameters" element="tns:SetCredentials" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/SetCredentials
```

The body of the SOAP message MUST contain the **SetCredentials** element.

##### 3.9.5.1.1.2   IDomainMaintenanceManager_SetCredentials_OutputMessage

This is the response for the **SetCredentials** operation.

```
<wsdl:message name="IDomainMaintenanceManager_SetCredentials_OutputMessage">
  <wsdl:part name="parameters" element="tns:SetCredentialsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/SetCredentialsResponse
```

The body of the SOAP message MUST contain the **SetCredentialsResponse** element.

### 3.9.5.1.2 Elements

#### 3.9.5.1.2.1 SetCredentials

This element specifies input values for the **SetCredentials** operation.

```
<xs:element name="SetCredentialsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="SetCredentialsResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.9.5.1.2.2 SetCredentialsResponse

This element specifies output values for the **SetCredentials** operation.

```
<xs:element name="SetCredentialsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="SetCredentialsResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.2 IDomainMaintenanceManager.ValidateCredentials

This operation validates the current user credentials.

```
<wsdl:operation name="ValidateCredentials">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/ValidateCredentials"
message="tns:IDomainMaintenanceManager_ValidateCredentials_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/ValidateCredentialsResponse"
message="tns:IDomainMaintenanceManager_ValidateCredentials_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/ValidateCredentialsDomainManagerFau
ltFault" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_ValidateCredentials_DomainManagerFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_ValidateCredentials_InputMessage** request message, the protocol server validates user credentials. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_ValidateCredentials_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.2.1 Messages

#### 3.9.5.2.1.1 IDomainMaintenanceManager_ValidateCredentials_InputMessage

This is the request for the **ValidateCredentials** operation.

```
<wsdl:message name="IDomainMaintenanceManager_ValidateCredentials_InputMessage">
  <wsdl:part name="parameters" element="tns:ValidateCredentials" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/ValidateCredentials
```

The body of the SOAP message MUST contain the **ValidateCredentials** element.

### 3.9.5.2.1.2    IDomainMaintenanceManager_ValidateCredentials_OutputMessage

This is the response for the **ValidateCredentials** operation.

```
<wsdl:message name="IDomainMaintenanceManager_ValidateCredentials_OutputMessage">
  <wsdl:part name="parameters" element="tns:ValidateCredentialsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/ValidateCredentialsResponse
```

The body of the SOAP message MUST contain the **ValidateCredentialsResponse** element.

### 3.9.5.2.2 Elements

### 3.9.5.2.2.1    ValidateCredentials

This element specifies input values for the **ValidateCredentials** operation.

```
<xs:element name="ValidateCredentials">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.9.5.2.2.2    ValidateCredentialsResponse

This element specifies output values for the **ValidateCredentials** operation.

```
<xs:element name="ValidateCredentialsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="ValidateCredentialsResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.3  IDomainMaintenanceManager.UpdateDnsRecords

This operation updates the DNS records for the domain.

```
<wsdl:operation name="UpdateDnsRecords">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecords"
message="tns:IDomainMaintenanceManager_UpdateDnsRecords_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecordsResponse"
message="tns:IDomainMaintenanceManager_UpdateDnsRecords_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecordsDomainManagerFaultF
ault" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_UpdateDnsRecords_DomainManagerFaultFault_FaultMessage"
/>
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecordsArgumentExceptionFa
ult" name="ArgumentExceptionFault"
message="tns:IDomainMaintenanceManager_UpdateDnsRecords_ArgumentExceptionFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_UpdateDnsRecords_InputMessage** request
message, the protocol server updates DNS records for the domain. Upon successful completion of the
action, the server MUST respond with
**IDomainMaintenanceManager_UpdateDnsRecords_OutputMessage**. In the event of a failure,
an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.3.1 Messages

#### 3.9.5.3.1.1    IDomainMaintenanceManager_UpdateDnsRecords_InputMessage

This is the request for the **UpdateDnsRecords** operation.

```
<wsdl:message name="IDomainMaintenanceManager_UpdateDnsRecords_InputMessage">
  <wsdl:part name="parameters" element="tns:UpdateDnsRecords" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecords
```

The body of the SOAP message MUST contain the **UpdateDnsRecords** element.

#### 3.9.5.3.1.2    IDomainMaintenanceManager_UpdateDnsRecords_OutputMessage

This is the response for the **UpdateDnsRecords** operation.

```
<wsdl:message name="IDomainMaintenanceManager_UpdateDnsRecords_OutputMessage">
  <wsdl:part name="parameters" element="tns:UpdateDnsRecordsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/UpdateDnsRecordsResponse
```

The body of the SOAP message MUST contain the **UpdateDnsRecordsResponse** element.

### 3.9.5.3.2 Elements

### 3.9.5.3.2.1 UpdateDnsRecords

This element specifies input values for the **UpdateDnsRecords** operation.

```
<xs:element name="UpdateDnsRecords">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="records" nillable="true"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q2:ArrayOfDnsRecord" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.3.2.2 UpdateDnsRecordsResponse

This element specifies output values for the **UpdateDnsRecords** operation.

```
<xs:element name="UpdateDnsRecordsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="UpdateDnsRecordsResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.4 IDomainMaintenanceManager.QueryAddressFromDNS

This operation returns the public IPv4 or IPv6 address corresponding to the currently configured domain name, using the public DNS infrastructure.

```
<wsdl:operation name="QueryAddressFromDNS">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromDNS"
message="tns:IDomainMaintenanceManager_QueryAddressFromDNS_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromDNSResponse"
message="tns:IDomainMaintenanceManager_QueryAddressFromDNS_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromDNSDomainManagerFau
ltFault" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_QueryAddressFromDNS_DomainManagerFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_QueryAddressFromDNS_InputMessage** request message, the protocol server determines the public IP address (IPv4 or IPv6) corresponding to the currently configured domain name using the public DNS infrastructure. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_QueryAddressFromDNS_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.4.1 Messages

### 3.9.5.4.1.1 IDomainMaintenanceManager_QueryAddressFromDNS_InputMessage

This is the request for the **QueryAddressFromDNS** operation.

```
<wsdl:message name="IDomainMaintenanceManager_QueryAddressFromDNS_InputMessage">
```

```
    <wsdl:part name="parameters" element="tns:QueryAddressFromDNS" />
  </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromDNS
```

The body of the SOAP message MUST contain the **QueryAddressFromDNS** element.

### 3.9.5.4.1.2    IDomainMaintenanceManager_QueryAddressFromDNS_OutputMessage

This is the response for the **QueryAddressFromDNS** operation.

```
<wsdl:message name="IDomainMaintenanceManager_QueryAddressFromDNS_OutputMessage">
  <wsdl:part name="parameters" element="tns:QueryAddressFromDNSResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromDNSResponse
```

The body of the SOAP message MUST contain the **QueryAddressFromDNSResponse** element.

### 3.9.5.4.2 Elements

### 3.9.5.4.2.1    QueryAddressFromDNS

This element specifies input values for the **QueryAddressFromDNS** operation.

```
<xs:element name="QueryAddressFromDNS">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.9.5.4.2.2    QueryAddressFromDNSResponse

This element specifies output values for the **QueryAddressFromDNS** operation.

```
<xs:element name="QueryAddressFromDNSResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="QueryAddressFromDNSResult" nillable="true"
xmlns:q3="http://schemas.datacontract.org/2004/07/System.Net" type="q3:IPAddress" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.5  IDomainMaintenanceManager.QueryAddressFromProvider

This operation returns the server's public IPv4 or IPv6 address by using the active maintenance provider.

```
<wsdl:operation name="QueryAddressFromProvider">
  <wsdl:input
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromProvider"
message="tns:IDomainMaintenanceManager_QueryAddressFromProvider_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromProviderResponse"
message="tns:IDomainMaintenanceManager_QueryAddressFromProvider_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromProviderDomainManag
erFaultFault" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_QueryAddressFromProvider_DomainManagerFaultFault_Fault
Message" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_QueryAddressFromProvider_InputMessage**
request message, the protocol server queries the active maintenance provider for the server's public
IP address. Upon successful completion of the action, the server MUST respond with
**IDomainMaintenanceManager_QueryAddressFromProvider_OutputMessage**. In the event of a
failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.5.1 Messages

### 3.9.5.5.1.1    IDomainMaintenanceManager_QueryAddressFromProvider_InputMessage

This is the request for the **QueryAddressFromProvider** operation.

```
<wsdl:message name="IDomainMaintenanceManager_QueryAddressFromProvider_InputMessage">
  <wsdl:part name="parameters" element="tns:QueryAddressFromProvider" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromProvider
```

The body of the SOAP message MUST contain the **QueryAddressFromProvider** element.

### 3.9.5.5.1.2    IDomainMaintenanceManager_QueryAddressFromProvider_OutputMessage

This is the response for the **QueryAddressFromProvider** operation.

```
<wsdl:message name="IDomainMaintenanceManager_QueryAddressFromProvider_OutputMessage">
  <wsdl:part name="parameters" element="tns:QueryAddressFromProviderResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/QueryAddressFromProviderResponse
```

The body of the SOAP message MUST contain the **QueryAddressFromProviderResponse** element.

### 3.9.5.5.2 Elements

### 3.9.5.5.2.1  QueryAddressFromProvider

This element specifies input values for the **QueryAddressFromProvider** operation.

```
<xs:element name="QueryAddressFromProvider">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.9.5.5.2.2  QueryAddressFromProviderResponse

This element specifies output values for the **QueryAddressFromProvider** operation.

```
<xs:element name="QueryAddressFromProviderResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="QueryAddressFromProviderResult" nillable="true"
xmlns:q4="http://schemas.datacontract.org/2004/07/System.Net" type="q4:IPAddress" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.9.5.6  IDomainMaintenanceManager.TestConnection

This operation returns a value indicating whether a connection to the current provider's backend service can be established.

```
<wsdl:operation name="TestConnection">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/TestConnection"
message="tns:IDomainMaintenanceManager_TestConnection_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/TestConnectionResponse"
message="tns:IDomainMaintenanceManager_TestConnection_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/TestConnectionDomainManagerFaultFau
lt" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_TestConnection_DomainManagerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_TestConnection_InputMessage** request message, the protocol server verifies whether a connection to the current provider's backend service can be established. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_TestConnection_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.6.1 Messages

### 3.9.5.6.1.1  IDomainMaintenanceManager_TestConnection_InputMessage

This is the request for the **TestConnection** operation.

```
<wsdl:message name="IDomainMaintenanceManager_TestConnection_InputMessage">
  <wsdl:part name="parameters" element="tns:TestConnection" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/TestConnection
```

The body of the SOAP message MUST contain the **TestConnection** element.

### 3.9.5.6.1.2   IDomainMaintenanceManager_TestConnection_OutputMessage

This is the response for the **TestConnection** operation.

```
<wsdl:message name="IDomainMaintenanceManager_TestConnection_OutputMessage">
  <wsdl:part name="parameters" element="tns:TestConnectionResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/TestConnectionResponse
```

The body of the SOAP message MUST contain the **TestConnectionResponse** element.

### 3.9.5.6.2 Elements

### 3.9.5.6.2.1   TestConnection

This element specifies input values for the **TestConnection** operation.

```
<xs:element name="TestConnection">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.9.5.6.2.2   TestConnectionResponse

This element specifies output values for the **TestConnection** operation.

```
<xs:element name="TestConnectionResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="TestConnectionResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.7  IDomainMaintenanceManager.GetDomainState

This operation returns the current domain state from the active maintenance provider.

```
<wsdl:operation name="GetDomainState">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDomainState"
message="tns:IDomainMaintenanceManager_GetDomainState_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDomainStateResponse"
message="tns:IDomainMaintenanceManager_GetDomainState_OutputMessage" />
```

```
      <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDomainStateDomainManagerFaultFau
lt" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_GetDomainState_DomainManagerFaultFault_FaultMessage"
/>
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_GetDomainState_InputMessage** request message, the protocol server queries the active maintenance provider for the current domain state. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_GetDomainState_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.7.1 Messages

#### 3.9.5.7.1.1    IDomainMaintenanceManager_GetDomainState_InputMessage

This is the request for the **GetDomainState** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDomainState_InputMessage">
  <wsdl:part name="parameters" element="tns:GetDomainState" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDomainState
```

The body of the SOAP message MUST contain the **GetDomainState** element.

#### 3.9.5.7.1.2    IDomainMaintenanceManager_GetDomainState_OutputMessage

This is the response for the **GetDomainState** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDomainState_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetDomainStateResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDomainStateResponse
```

The body of the SOAP message MUST contain the **GetDomainStateResponse** element.

### 3.9.5.7.2 Elements

#### 3.9.5.7.2.1    GetDomainState

This element specifies input values for the **GetDomainState** operation.

```
<xs:element name="GetDomainState">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
```

```
</xs:element>
```

### 3.9.5.7.2.2   GetDomainStateResponse

This element specifies output values for the **GetDomainState** operation.

```
<xs:element name="GetDomainStateResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetDomainStateResult" nillable="true"
xmlns:q5="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q5:DomainState" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.9.5.8  IDomainMaintenanceManager.GetCertificateState

This operation returns the current certificate state from the active maintenance provider.

```
<wsdl:operation name="GetCertificateState">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetCertificateState"
message="tns:IDomainMaintenanceManager_GetCertificateState_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetCertificateStateResponse"
message="tns:IDomainMaintenanceManager_GetCertificateState_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetCertificateStateDomainManagerFau
ltFault" name="DomainManagerFaultFault"
message="tns:IDomainMaintenanceManager_GetCertificateState_DomainManagerFaultFault_FaultMessa
ge" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_GetCertificateState_InputMessage** request
message, the protocol server queries the active maintenance provider for the current certificate state.
Upon successful completion of the action, the server MUST respond with
**IDomainMaintenanceManager_GetCertificateState_OutputMessage**. In the event of a failure,
an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.8.1 Messages

#### 3.9.5.8.1.1   IDomainMaintenanceManager_GetCertificateState_InputMessage

This is the request for the **GetCertificateState** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetCertificateState_InputMessage">
  <wsdl:part name="parameters" element="tns:GetCertificateState" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetCertificateState
```

The body of the SOAP message MUST contain the **GetCertificateState** element.

#### 3.9.5.8.1.2   IDomainMaintenanceManager_GetCertificateState_OutputMessage

This is the response for the **GetCertificateState** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetCertificateState_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetCertificateStateResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetCertificateStateResponse
```

The body of the SOAP message MUST contain the **GetCertificateStateResponse** element.

### 3.9.5.8.2 Elements

#### 3.9.5.8.2.1    GetCertificateState

This element specifies input values for the **GetCertificateState** operation.

```
<xs:element name="GetCertificateState">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.9.5.8.2.2    GetCertificateStateResponse

This element specifies output values for the **GetCertificateState** operation.

```
<xs:element name="GetCertificateStateResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetCertificateStateResult" nillable="true"
xmlns:q6="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q6:CertificateState" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.9  IDomainMaintenanceManager.GetDomainNameConfiguration

This operation returns the currently active domain name configuration.

```
<wsdl:operation name="GetDomainNameConfiguration">
  <wsdl:input
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDomainNameConfiguration"
message="tns:IDomainMaintenanceManager_GetDomainNameConfiguration_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDomainNameConfigurationResponse"
message="tns:IDomainMaintenanceManager_GetDomainNameConfiguration_OutputMessage" />
</wsdl:operation>
```

Upon receiving the
**IDomainMaintenanceManager_GetDomainNameConfiguration_InputMessage** request
message, the protocol server returns the currently active domain name configuration. Upon successful
completion of the action, the server MUST respond with

**IDomainMaintenanceManager_GetDomainNameConfiguration_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.9.1 Messages

#### 3.9.5.9.1.1    IDomainMaintenanceManager_GetDomainNameConfiguration_InputMessage

This is the request for the **GetDomainNameConfiguration** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDomainNameConfiguration_InputMessage">
  <wsdl:part name="parameters" element="tns:GetDomainNameConfiguration" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDomainNameConfiguration
```

The body of the SOAP message MUST contain the **GetDomainNameConfiguration** element.

#### 3.9.5.9.1.2    IDomainMaintenanceManager_GetDomainNameConfiguration_OutputMessage

This is the response for the **GetDomainNameConfiguration** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDomainNameConfiguration_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetDomainNameConfigurationResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDomainNameConfigurationResponse
```

The body of the SOAP message MUST contain the **GetDomainNameConfigurationResponse** element.

### 3.9.5.9.2 Elements

#### 3.9.5.9.2.1    GetDomainNameConfiguration

This element specifies input values for the **GetDomainNameConfiguration** operation.

```
<xs:element name="GetDomainNameConfiguration">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.9.5.9.2.2    GetDomainNameConfigurationResponse

This element specifies output values for the **GetDomainNameConfiguration** operation.

```
<xs:element name="GetDomainNameConfigurationResponse">
```

```
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="GetDomainNameConfigurationResult" nillable="true"
xmlns:q7="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q7:DomainNameConfiguration" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
```

### 3.9.5.10    IDomainMaintenanceManager.GetDynamicDNSInfo

This operation returns data about dynamic DNS updates, for example, the time of the most recent successful update.

```
<wsdl:operation name="GetDynamicDNSInfo">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDynamicDNSInfo"
message="tns:IDomainMaintenanceManager_GetDynamicDNSInfo_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetDynamicDNSInfoResponse"
message="tns:IDomainMaintenanceManager_GetDynamicDNSInfo_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_GetDynamicDNSInfo_InputMessage** request message, the protocol server returns data about dynamic DNS updates (the time of the last most recent successful update, for example). Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_GetDynamicDNSInfo_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

#### 3.9.5.10.1    Messages

#### 3.9.5.10.1.1 IDomainMaintenanceManager_GetDynamicDNSInfo_InputMessage

This is the request for the **GetDynamicDNSInfo** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDynamicDNSInfo_InputMessage">
  <wsdl:part name="parameters" element="tns:GetDynamicDNSInfo" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDynamicDNSInfo
```

The body of the SOAP message MUST contain the **GetDynamicDNSInfo** element.

#### 3.9.5.10.1.2 IDomainMaintenanceManager_GetDynamicDNSInfo_OutputMessage

This is the response for the **GetDynamicDNSInfo** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetDynamicDNSInfo_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetDynamicDNSInfoResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IDomainMaintenanceManager/GetDynamicDNSInfoResponse
```

The body of the SOAP message MUST contain the **GetDynamicDNSInfoResponse** element.

### 3.9.5.10.2    Elements

#### 3.9.5.10.2.1  GetDynamicDNSInfo

This element specifies input values for the **GetDynamicDNSInfo** operation.

```
<xs:element name="GetDynamicDNSInfo">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

#### 3.9.5.10.2.2  GetDynamicDNSInfoResponse

This element specifies output values for the **GetDynamicDNSInfo** operation.

```
<xs:element name="GetDynamicDNSInfoResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetDynamicDNSInfoResult" nillable="true"
xmlns:q8="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q8:DynamicDNSInfo" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.9.5.11        IDomainMaintenanceManager.GetSettings

This operation returns the current domain manager settings.

```
<wsdl:operation name="GetSettings">
  <wsdl:input wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetSettings"
message="tns:IDomainMaintenanceManager_GetSettings_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IDomainMaintenanceManager/GetSettingsResponse"
message="tns:IDomainMaintenanceManager_GetSettings_OutputMessage" />
</wsdl:operation>
```

Upon receiving the **IDomainMaintenanceManager_GetSettings_InputMessage** request message, the protocol server returns the current domain manager settings. Upon successful completion of the action, the server MUST respond with **IDomainMaintenanceManager_GetSettings_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.9.5.11.1    Messages

#### 3.9.5.11.1.1  IDomainMaintenanceManager_GetSettings_InputMessage

This is the request for the **GetSettings** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetSettings_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSettings" />
```

```
        </wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IDomainMaintenanceManager/GetSettings
```

The body of the SOAP message MUST contain the **GetSettings** element.

### 3.9.5.11.1.2 IDomainMaintenanceManager_GetSettings_OutputMessage

This is the response for the **GetSettings** operation.

```
<wsdl:message name="IDomainMaintenanceManager_GetSettings_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSettingsResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
    http://tempuri.org/IDomainMaintenanceManager/GetSettingsResponse
```

The body of the SOAP message MUST contain the **GetSettingsResponse** element.

### 3.9.5.11.2 Elements

### 3.9.5.11.2.1 GetSettings

This element specifies input values for the **GetSettings** operation.

```
<xs:element name="GetSettings">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.9.5.11.2.2 GetSettingsResponse

This element specifies output values for the **GetSettings** operation.

```
<xs:element name="GetSettingsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetSettingsResult" nillable="true"
xmlns:q9="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.RemoteAcce
ss.Domains" type="q9:DomainProviderManagerSettings" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

## 3.9.6 Timer Events

None.

### 3.9.7 Other Local Events

None.

## 3.10 AzureAD and HostedEmail Integration Service Contract Details

None.

### 3.10.1 Abstract Data Model

None.

### 3.10.2 Timers

None.

### 3.10.3 Initialization

None.

### 3.10.4 Higher-Layer Triggered Events

None

### 3.10.5 Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IPasswordSyncClientProvider.Disconnect | Disconnects from the provider service. |
| IPasswordSyncClientProvider.RaiseOperationInvokeEvent | Raises an operation invoke event. |
| IO365ClientProvider.GetSharePointAddress<19> | Returns the address of the SharePoint web portal associated with the Microsoft Office 365 subscription. |
| IO365ClientProvider.IsActivated | Checks whether Office 365 integration is activated. |
| IPasswordSyncClientProvider.GetPasswordChangeStatus | Gets password change status. |
| IPasswordSyncClientProvider.MarkPasswordChangeFixed | Marks password change status as fixed. |
| IO365ClientProvider.GetEmailAddress | Gets the email address for the caller. |

#### 3.10.5.1    IPasswordSyncClientProvider.Disconnect

This operation disconnects from the provider service.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="true" name="Disconnect">
  <wsdl:input wsam:Action="http://tempuri.org/IProvider/Disconnect" message="tns:
IPasswordSyncClientProvider_Disconnect_InputMessage" />
</wsdl:operation>
```

Upon receiving the **IPasswordSyncClientProvider_Disconnect_InputMessage** request message, the protocol server disconnects the caller from the provider service. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.1.1    Messages

#### 3.10.5.1.1.1  IPasswordSyncClientProvider_Disconnect_InputMessage

This is the request for the **Disconnect** operation.

```
<wsdl:message name="IPasswordSyncClientProvider_Disconnect_InputMessage">
  <wsdl:part name="parameters" element="tns:Disconnect" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/Disconnect
```

The body of the SOAP message MUST contain the **Disconnect** element.

### 3.10.5.1.2    Elements

#### 3.10.5.1.2.1  Disconnect

This element specifies input values for the **Disconnect** operation.

```
<xs:element name="Disconnect">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

## 3.10.5.2        IPasswordSyncClientProvider.RaiseOperationInvokeEvent

This operation raises an operation invoke event.<20>

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false"
name="RaiseOperationInvokeEvent">
  <wsdl:output wsam:Action="http://tempuri.org/IProvider/RaiseOperationInvokeEvent"
message="tns: IPasswordSyncClientProvider_RaiseOperationInvokeEvent_OutputCallbackMessage" />
</wsdl:operation>
```

### 3.10.5.2.1    Messages

#### 3.10.5.2.1.1  IO365ClientProvider_RaiseOperationInvokeEvent_OutputCallbackMessage

This is the request for the **RaiseOperationInvoke** operation.

```
<wsdl:message name="IO365ClientProvider_RaiseOperationInvokeEvent_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:RaiseOperationInvokeEvent" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/RaiseOperationInvokeEvent
```

The body of the SOAP message MUST contain the **RaiseOperationInvokeEvent** element.

### 3.10.5.2.2    Elements

### 3.10.5.2.2.1  RaiseOperationInvokeEvent

This element specifies input values for the **RaiseOperationInvoke** operation.

```
<xs:element name="RaiseOperationInvokeEvent">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="operationName" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="result" nillable="true" type="xs:anyType" />
      <xs:element minOccurs="0" name="errorException" nillable="true"
xmlns:q1="http://schemas.datacontract.org/2004/07/System" type="q1:Exception" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.3        IO365ClientProvider.GetSharePointAddress

This operation MAY<21> return the address of the SharePoint web portal associated with the Office 365 subscription.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false"
name="GetSharePointAddress">
  <wsdl:input wsam:Action="http://tempuri.org/IO365ClientProvider/GetSharePointAddress"
message="tns:IO365ClientProvider_GetSharePointAddress_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IO365ClientProvider/GetSharePointAddressResponse"
message="tns:IO365ClientProvider_GetSharePointAddress_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IO365ClientProvider/GetSharePointAddressO365ConfigureExceptio
nFault" name="O365ConfigureExceptionFault"
message="tns:IO365ClientProvider_GetSharePointAddress_O365ConfigureExceptionFault_FaultMessag
e" />
</wsdl:operation>
```

Upon receiving the **IO365ClientProvider_GetSharePointAddress_InputMessage** request message, the protocol server returns the SharePoint address. Upon successful completion of the action, the server MUST respond with **IO365ClientProvider_GetSharePointAddress_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.3.1    Messages

### 3.10.5.3.1.1  IO365ClientProvider_GetSharePointAddress_InputMessage

This is the request for the **GetSharePointAddress** operation.

```
<wsdl:message name="IO365ClientProvider_GetSharePointAddress_InputMessage">
  <wsdl:part name="parameters" element="tns:GetSharePointAddress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetSharePointAddress
```

The body of the SOAP message MUST contain the **GetSharePointAddress** element.

### 3.10.5.3.1.2  IO365ClientProvider_GetSharePointAddress_OutputMessage

This is the response for the **GetSharePointAddress** operation.

```
<wsdl:message name="IO365ClientProvider_GetSharePointAddress_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetSharePointAddressResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetSharePointAddressResponse
```

The body of the SOAP message MUST contain the **GetSharePointAddressResponse** element.

### 3.10.5.3.2    Elements

### 3.10.5.3.2.1  GetSharePointAddress

This element specifies input values for the **GetSharePointAddress** operation.

```
<xs:element name="GetSharePointAddress">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.10.5.3.2.2  GetSharePointAddressResponse

This element specifies output values for the **GetSharePointAddress** operation.

```
<xs:element name="GetSharePointAddressResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetSharePointAddressResult" nillable="true"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.4      IO365ClientProvider.IsActivated

This operation MAY<22> return a value indicating whether Office 365 integration is activated.

```
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="IsActivated">
  <wsdl:input wsam:Action="http://tempuri.org/IO365ClientProvider/IsActivated"
message="tns:IO365ClientProvider_IsActivated_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IO365ClientProvider/IsActivatedResponse"
message="tns:IO365ClientProvider_IsActivated_OutputMessage" />
```

```
      <wsdl:fault
wsam:Action="http://tempuri.org/IO365ClientProvider/IsActivatedO365ConfigureExceptionFault"
name="O365ConfigureExceptionFault"
message="tns:IO365ClientProvider_IsActivated_O365ConfigureExceptionFault_FaultMessage" />
   </wsdl:operation>
```

Upon receiving the **IO365ClientProvider_IsActivated_InputMessage** request message, the protocol server checks whether the Office 365 integration is activated. Upon successful completion of the action, the server MUST respond with **IO365ClientProvider_IsActivated_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.4.1    Messages

#### 3.10.5.4.1.1  IO365ClientProvider_IsActivated_InputMessage

This is the request for the **IsActivated** operation.

```
<wsdl:message name="IO365ClientProvider_IsActivated_InputMessage">
  <wsdl:part name="parameters" element="tns:IsActivated" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/IsActivated
```

The body of the SOAP message MUST contain the **IsActivated** element.

#### 3.10.5.4.1.2  IO365ClientProvider_IsActivated_OutputMessage

This is the response for the **IsActivated** operation.

```
<wsdl:message name="IO365ClientProvider_IsActivated_OutputMessage">
  <wsdl:part name="parameters" element="tns:IsActivatedResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/IsActivatedResponse
```

The body of the SOAP message MUST contain the **IsActivatedResponse** element.

### 3.10.5.4.2    Elements

#### 3.10.5.4.2.1  IsActivated

This element specifies input values for the **IsActivated** operation.

```
<xs:element name="IsActivated">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.10.5.4.2.2  IsActivatedResponse

This element specifies output values for the **IsActivated** operation.

```
<xs:element name="IsActivatedResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="IsActivatedResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.5        IPasswordSyncClientProvider.GetPasswordChangeStatus

This operation returns the password change status.

```
name="GetPasswordChangeStatus">
  <wsdl:input
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/GetPasswordChangeStatus"
message="tns:IPasswordSyncClientProvider_GetPasswordChangeStatus_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/GetPasswordChangeStatusResponse"
message="tns:IPasswordSyncClientProvider_GetPasswordChangeStatus_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/GetPasswordChangeStatusPwdSyncCon
figureExceptionFault" name="PwdSyncConfigureExceptionFault"
message="tns:IPasswordSyncClientProvider_GetPasswordChangeStatus_PwdSyncConfigureExceptionFau
lt_FaultMessage" />
```

Upon receiving the **IPasswordSyncClientProvider_GetPasswordChangeStatus_InputMessage** request message, the protocol server returns the password change status. Upon successful completion of the action, the server MUST respond with **IPasswordSyncClientProvider_GetPasswordChangeStatus_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.5.1    Messages

### 3.10.5.5.1.1  IPasswordSyncClientProvider_GetPasswordChangeStatus_InputMessage

This is the request for the **GetPasswordChangeStatus** operation.

```
<wsdl:message name="IPasswordSyncClientProvider_GetPasswordChangeStatus_InputMessage">
  <wsdl:part name="parameters" element="tns:GetPasswordChangeStatus" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetPasswordChangeStatus
```

The body of the SOAP message MUST contain the **GetPasswordChangeStatus** element.

### 3.10.5.5.1.2  IPasswordSyncClientProvider_GetPasswordChangeStatus_OutputMessage

This is the response for the **GetPasswordChangeStatus** operation.

```
<wsdl:message name="IPasswordSyncClientProvider_GetPasswordChangeStatus_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetPasswordChangeStatusResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetPasswordChangeStatusResponse
```

The body of the SOAP message MUST contain the **GetPasswordChangeStatusResponse** element.

### 3.10.5.5.2    Elements

### 3.10.5.5.2.1  GetPasswordChangeStatus

This element specifies input values for the **GetPasswordChangeStatus** operation.

```
<xs:element name="GetPasswordChangeStatus">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="localuser" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.5.2.2  GetPasswordChangeStatusResponse

This element specifies output values for the **GetPasswordChangeStatus** operation.

```
<xs:element name="GetPasswordChangeStatusResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetPasswordChangeStatusResult"
xmlns:q2="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.
PasswordSync " type="q3:PasswordChangeStatus" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.6    IPasswordSyncClientProvider.MarkUserPasswordChangeFixed

This operation marks the password change status as fixed.

```
</wsdl:operation>
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false"
name="MarkPasswordChangeFixed">
  <wsdl:input
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/MarkUserPasswordChangeFixed"
message="tns:IPasswordSyncClientProvider_MarkPasswordChangeFixed_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/MarkUserPasswordChangeFixedRespon
se" message="tns:IPasswordSyncClientProvider_MarkPasswordChangeFixed_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IPasswordSyncClientProvider/MarkUserPasswordChangeFixedPwdSyn
cConfigureExceptionFault" name="PwdSyncConfigureExceptionFault"
message="tns:IPasswordSyncClientProvider_MarkUserPasswordChangeFixed_PwdSyncConfigureExceptio
nFault_FaultMessage" />
```

Upon receiving the **IPasswordSyncClientProvider_MarkPasswordChangeFixed_InputMessage** request message, the protocol server marks the password change status as fixed. Upon successful completion of the action, the server MUST respond with **IPasswordSyncClientProvider_MarkPasswordChangeFixed_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.6.1    Messages

#### 3.10.5.6.1.1  IPasswordSyncClientProvider_MarkUserPasswordChangeFixed_InputMessage

This is the request for the **MarkPasswordChangeFixed** operation.

```
<wsdl:message name="IPasswordSyncClientProvider_MarkUserPasswordChangeFixed_InputMessage">
  <wsdl:part name="parameters" element="tns:MarkPasswordChangeFixed" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/MarkUserPasswordChangeFixed
```

The body of the SOAP message MUST contain the **MarkUserPasswordChangeFixed** element.

#### 3.10.5.6.1.2  IPasswordSyncClientProvider_MarkUserPasswordChangeFixed_OutputMessage

This is the response for the **MarkPasswordChangeFixed** operation.

```
<wsdl:message name="IPasswordSyncClientProvider_MarkPasswordChangeFixed_OutputMessage">
  <wsdl:part name="parameters" element="tns:MarkPasswordChangeFixedResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/MarkUserPasswordChangeFixedResponse
```

The body of the SOAP message MUST contain the **MarkUserPasswordChangeFixedResponse** element.

### 3.10.5.6.2    Elements

#### 3.10.5.6.2.1  MarkUserPasswordChangeFixed

This element specifies input values for the **MarkUserPasswordChangeFixed** operation.

```
<xs:element name="MarkUserPasswordChangeFixed">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="localuser" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.5.6.2.2 MarkUserPasswordChangeFixedResponse

This element specifies output values for the **MarkUserPasswordChangeFixed** operation.

```
<xs:element name="MarkUserPasswordChangeFixedResponse">
  <xs:complexType>
    <xs:sequence />
  </xs:complexType>
</xs:element>
```

### 3.10.5.7    IO365ClientProvider.GetEmailAddress

This operation MAY<23> return the email address for the caller.

```
</wsdl:operation>
<wsdl:operation msc:isInitiating="true" msc:isTerminating="false" name="GetEmailAddress">
  <wsdl:input wsam:Action="http://tempuri.org/IO365ClientProvider/GetEmailAddress"
message="tns:IO365ClientProvider_GetEmailAddress_InputMessage" />
  <wsdl:output wsam:Action="http://tempuri.org/IO365ClientProvider/GetEmailAddressResponse"
message="tns:IO365ClientProvider_GetEmailAddress_OutputMessage" />
  <wsdl:fault
wsam:Action="http://tempuri.org/IO365ClientProvider/GetEmailAddressO365ConfigureExceptionFaul
t" name="O365ConfigureExceptionFault"
message="tns:IO365ClientProvider_GetEmailAddress_O365ConfigureExceptionFault_FaultMessage" />
</wsdl:operation>
```

Upon receiving the **IO365ClientProvider_GetEmailAddress_InputMessage** request message, the protocol server marks the password change status as fixed. Upon successful completion of the action, the server MUST respond with **IO365ClientProvider_GetEmailAddress_OutputMessage**. In the event of a failure, an appropriate SOAP fault MUST be sent to the client as specified in section 2.2.2.1.

### 3.10.5.7.1    Messages

### 3.10.5.7.1.1 IO365ClientProvider_GetEmailAddress_InputMessage

This is the request for the **GetEmailAddress** operation.

```
<wsdl:message name="IO365ClientProvider_GetEmailAddress_InputMessage">
  <wsdl:part name="parameters" element="tns:GetEmailAddress" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetEmailAddress
```

The body of the SOAP message MUST contain the **GetEmailAddress** element.

### 3.10.5.7.1.2 IO365ClientProvider_GetEmailAddress_OutputMessage

This is the response for the **GetEmailAddress** operation.

```
<wsdl:message name="IO365ClientProvider_GetEmailAddress_OutputMessage">
  <wsdl:part name="parameters" element="tns:GetEmailAddressResponse" />
</wsdl:message>
```

This message MUST be sent with the following SOAP action.

```
http://tempuri.org/IProvider/GetEmailAddressResponse
```

The body of the SOAP message MUST contain the **GetEmailAddressResponse** element.

### 3.10.5.7.2   Elements

#### 3.10.5.7.2.1  GetEmailAddress

This element specifies input values for the **GetEmailAddress** operation.

```
<xs:element name="GetEmailAddress">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="localuser" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

#### 3.10.5.7.2.2  GetEmailAddressResponse

This element specifies output values for the **GetEmailAddress** operation.

```
<xs:element name="GetEmailAddressResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetEmailAddressResult" nillable="true" type="xs:string"
/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.10.6 Timer Events

None.

### 3.10.7 Other Local Events

None.

## 3.11  Client Notification Service Contract Details

### 3.11.1 Abstract Data Model

None.<24>

### 3.11.2 Timers

None.

### 3.11.3 Initialization

None.

## 3.11.4 Higher-Layer Triggered Events

None.

## 3.11.5 Message Processing Events and Sequencing Rules

| OperationContract | Description |
|---|---|
| IClientRegistrationService.RegisterClient | Registers a client callback channel to the server. |
| IClientNotificationCallback.NotificationReceived | Indicates a notification event has been received from server. |
| IClientNotificationProvider.NotifyClientByName | Sends a notification to a client with the specified machine name. |
| IClientNotificationProvider.NotifyClientById | Sends a notification to a client with the specified SID. |
| IClientNotificationProvider.NotifyAllClients | Sends a notification to all online clients. |

### 3.11.5.1          IClientRegistrationService.RegisterClient

The operation registers the client callback channel to the server.

```
<wsdl:operation name="RegisterClient">
  <wsdl:input wsam:Action="http://tempuri.org/IClientRegistrationService/RegisterClient"
message="tns:IClientRegistrationService_RegisterClient_InputMessage" />
  </wsdl:operation>
```

Upon receiving the **IClientRegistrationService_RegisterClient_InputMessage** request message, the protocol server adds the callback channel handle to its dictionary and the server, the protocol server MAY send notification messages through this channel.

#### 3.11.5.1.1       Messages

#### 3.11.5.1.1.1 IClientRegistrationService_RegisterClient_InputMessage

This is the request for the **RegisterClient** operation.

```
<wsdl:message name="IClientRegistrationService_RegisterClient_InputMessage">
  <wsdl:part name="parameters" element="tns:RegisterClient" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action.

```
http://tempuri.org/IClientRegistrationService/RegisterClient
```

The body of the SOAP message MUST contain the **RegisterClient** element.

#### 3.11.5.1.2       Elements

#### 3.11.5.1.2.1 RegisterClient

This element specifies input values for the RegisterClient operation.

```
<xs:element name="RegisterClient">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="clientId" nillable="true" type="xs:string" />
      </xs:sequence>
    </xs:complexType>
</xs:element>
```

## 3.11.5.2 IClientNotificationCallback.NotificationReceived

The operation is a callback that processes a notification message from the server.

```
<wsdl:operation name="NotificationReceived">
  <wsdl:output
wsam:Action="http://tempuri.org/IClientRegistrationService/NotificationReceived"
message="tns:IClientRegistrationService_NotificationReceived_OutputCallbackMessage" />
  <wsdl:input
wsam:Action="http://tempuri.org/IClientRegistrationService/NotificationReceivedResponse"
message="tns:IClientRegistrationService_NotificationReceived_InputCallbackMessage" />
</wsdl:operation>
```

### 3.11.5.2.1 Messages

#### 3.11.5.2.1.1 IClientNotificationCallback_NotificationReceived_OutputCallbackMessage

This message is the request for the **NotificationReceived** operation.

```
<wsdl:message name="IClientRegistrationService_NotificationReceived_OutputCallbackMessage">
  <wsdl:part name="parameters" element="tns:NotificationReceived" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
http://tempuri.org/IClientRegistrationService/NotificationReceived
```

The body of the SOAP message MUST contain the **NotificationReceived** element.

#### 3.11.5.2.1.2 IClientNotificationCallback_NotificationReceived_InputCallbackMessage

This message is the response for the **NotificationReceived** operation.

```
<wsdl:message name="IClientRegistrationService_NotificationReceived_InputCallbackMessage">
  <wsdl:part name="parameters" element="tns:NotificationReceivedResponse" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
http://tempuri.org/IClientRegistrationService/NotificationReceived
```

The body of the SOAP message MUST contain the **NotificationReceivedResponse** element.

### 3.11.5.2.2 Elements

### 3.11.5.2.2.1 NotificationReceived

This element specifies input values for the NotificationReceived operation.

```
<xs:element name="NotificationReceived">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="messageId" type="xs:unsignedInt" />
      <xs:element minOccurs="0" name="message" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.11.5.2.2.2 NotificationReceivedResponse

This element specifies input values for the NotificationReceivedResponse operation.

```
<xs:element name="NotificationReceivedResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="NotificationReceivedResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.11.5.3      IClientNotificationProvider.NotifyClientByName

The operation sends a notification message to a client by using the machine name.

```
    <wsdl:operation name="NotifyClientByName">
      <wsdl:input
wsam:Action="http://tempuri.org/IClientNotificationProvider/NotifyClientByName"
message="tns:IClientNotificationProvider_NotifyClientByName_InputMessage" />
      <wsdl:output
wsam:Action="http://tempuri.org/ClientNotificationProvider/NotifyClientByNameResponse"
message="tns:IClientNotificationProvider_NotifyClientByName_OutputMessage" />
    </wsdl:operation>
```

### 3.11.5.3.1    Messages

### 3.11.5.3.1.1  IClientNotificationProvider_NotifyClientByName_InputMessage

This message is the request for the **NotifyClientByName** operation.

```
    <wsdl:message name="IClientNotificationProvider_NotifyClientByName_InputMessage">
      <wsdl:part name="parameters" element="tns:NotifyClientByName" />
    </wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
    http://tempuri.org/IClientNotificationProvider/NotifyClientByName
```

The body of the SOAP message MUST contain the **NotifyClientByName** element.

### 3.11.5.3.1.2  IClientNotificationProvider_NotifyClientByName_OutputMessage

This message is the response for the **NotifyClientByName** operation.

```
<wsdl:message name="IClientNotificationProvider_NotifyClientByName_OutputMessage">
  <wsdl:part name="parameters" element="tns:NotifyClientByNameResponse" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
http://tempuri.org/ClientNotificationProvider/NotifyClientByNameResponse
```

The body of the SOAP message MUST contain the **NotifyClientByNameResponse** element.

### 3.11.5.3.2     Elements

### 3.11.5.3.2.1  NotifyClientByName

This element specifies input values for the **NotifyClientByName** operation.

```
<xs:element name="NotifyClientByName">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="name" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="messageId" type="xs:unsignedInt" />
      <xs:element minOccurs="0" name="message" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.11.5.3.2.2  NotifyClientByNameResponse

This element specifies output values for the **NotifyClientByName** operation.

```
<xs:element name="NotifyClientByNameResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="NotifyClientByNameResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.11.5.4        IClientNotificationProvider.NotifyClientById

The operation sends a notification message to a client by using the machine SID.

```
<wsdl:operation name="NotifyClientById">
  <wsdl:input
wsam:Action="http://tempuri.org/IClientNotificationProvider/NotifyClientById"
message="tns:IClientNotificationProvider_NotifyClientById_InputMessage" />
  <wsdl:output
wsam:Action="http://tempuri.org/ClientNotificationProvider/NotifyClientByIdResponse"
message="tns:IClientNotificationProvider_NotifyClientById_OutputMessage" />
</wsdl:operation>
```

### 3.11.5.4.1     Messages

### 3.11.5.4.1.1 IClientNotificationProvider_NotifyClientById_InputMessage

This message is the request for the **NotifyClientById** operation.

```
<wsdl:message name="IClientNotificationProvider_NotifyClientById_InputMessage">
  <wsdl:part name="parameters" element="tns:NotifyClientById" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
http://tempuri.org/IClientNotificationProvider/NotifyClientById
```

The body of the SOAP message MUST contain the **NotifyClientById** element.

### 3.11.5.4.1.2 IClientNotificationProvider_NotifyClientById_OutputMessage

This message is the response for the **NotifyClientById** operation.

```
<wsdl:message name="IClientNotificationProvider_NotifyClientById_OutputMessage">
  <wsdl:part name="parameters" element="tns:NotifyClientByIdResponse" />
</wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
http://tempuri.org/ClientNotificationProvider/NotifyClientByIdResponse
```

The body of the SOAP message MUST contain the **NotifyClientByIdResponse** element.

### 3.11.5.4.2    Elements

### 3.11.5.4.2.1  NotifyClientById

This element specifies input values for the **NotifyClientById** operation.

```
<xs:element name="NotifyClientById">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="id" nillable="true" type="xs:string" />
      <xs:element minOccurs="0" name="messageId" type="xs:unsignedInt" />
      <xs:element minOccurs="0" name="message" nillable="true" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### 3.11.5.4.2.2  NotifyClientByIdResponse

This element specifies output values for the **NotifyClientById** operation.

```
<xs:element name="NotifyClientByIdResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="NotifyClientByIdResult" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
```

```
        </xs:element>
```

### 3.11.5.5 IClientNotificationProvider.NotifyAllClients

The operation sends notification messages to all the online clients.

```
    <wsdl:operation name="NotifyAllClients">
      <wsdl:input
wsam:Action="http://tempuri.org/IClientNotificationProvider/NotifyAllClients"
message="tns:IClientNotificationProvider_NotifyAllClients_InputMessage" />
      <wsdl:output
wsam:Action="http://tempuri.org/ClientNotificationProvider/NotifyAllClientsResponse"
message="tns:IClientNotificationProvider_NotifyAllClients_OutputMessage" />
    </wsdl:operation>
```

#### 3.11.5.5.1 Messages

##### 3.11.5.5.1.1 IClientNotificationProvider_NotifyAllClients_InputMessage

This message is the request for the **NotifyAllClients** operation.

```
    <wsdl:message name="IClientNotificationProvider_NotifyAllClients_InputMessage">
      <wsdl:part name="parameters" element="tns:NotifyAllClients" />
    </wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
    http://tempuri.org/IClientNotificationProvider/NotifyAllClients
```

The body of the SOAP message MUST contain the **NotifyAllClients** element.

##### 3.11.5.5.1.2 IClientNotificationProvider_NotifyAllClients_OutputMessage

This message is the response for the **NotifyAllClients** operation.

```
    <wsdl:message name="IClientNotificationProvider_NotifyAllClients_OutputMessage">
      <wsdl:part name="parameters" element="tns:NotifyAllClientsResponse" />
    </wsdl:message>
```

The message MUST be sent with the following SOAP action:

```
    http://tempuri.org/ClientNotificationProvider/NotifyAllClientsResponse
```

The body of the SOAP message MUST contain the **NotifyAllClientsResponse** element.

#### 3.11.5.5.2 Elements

##### 3.11.5.5.2.1 NotifyAllClients

This element specifies input values for the **NotifyAllClients** operation.

```
    <xs:element name="NotifyAllClients">
      <xs:complexType>
```

```
<xs:sequence>
    <xs:element minOccurs="0" name="messageId" type="xs:unsignedInt" />
    <xs:element minOccurs="0" name="message" nillable="true" type="xs:string" />
</xs:sequence>
</xs:complexType>
</xs:element>
```

### 3.11.5.5.2.2  NotifyAllClientsResponse

This element specifies output values for the **NotifyAllClients** operation.

```
<xs:element name="NotifyAllClientsResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="NotifyAllClientsResult" type="NotificationResults" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

# 4 Protocol Examples

## 4.1 Get the Computer Backup Service Status

The following is the SOAP request message that can be sent to get the status of Windows Server operating system Client Computer Backup Service.

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IPCBackupServerProvider/GetServiceStatus</a:Action>
    <a:MessageID>urn:uuid:18c442b8-7715-4895-ae63-1d036d073091</a:MessageID>
    <a:ReplyTo>
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
    <a:To s:mustUnderstand="1">net.tcp://liming-
sbs1:6602/Microsoft.WindowsServerSolutions.wssg_pc_backup_server/Microsoft.WindowsServerSolut
ions.DataProtection.PCBackup.BackupUtil.WcfContracts.IPCBackupServerProvider</a:To>
  </s:Header>
  <s:Body>
    <GetServiceStatus xmlns="http://tempuri.org/"></GetServiceStatus>
  </s:Body>
</s:Envelope>
```

The following is one example of the SOAP response message for the above request.

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IPCBackupServerProvider/GetServiceStatusResponse</a:A
ction>
  </s:Header>
  <s:Body>
    <GetServiceStatusResponse xmlns="http://tempuri.org/">
      <GetServiceStatusResult
xmlns:d4p1="http://schemas.datacontract.org/2004/07/Microsoft.WindowsServerSolutions.DataProt
ection.PCBackup.BackupUtil.WcfContracts" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4p1:Ready>true</d4p1:Ready>
      </GetServiceStatusResult>
    </GetServiceStatusResponse>
  </s:Body>
</s:Envelope>
```

# 5   Security

## 5.1   Security Considerations for Implementers

The Provider Framework protocol ~~should~~is recommended to use TCP transport security in order to secure the communication between protocol client and protocol server, as well as to support authentication. The protocol ~~should~~is recommended to support two authentication types:

1. By user identity

2. By machine identity

Each provider can specify its appropriate authentication type.

### 5.1.1   User Identity

Stream security settings ~~should~~are recommended to be bound to the endpoint, with protection level configured to support encryption and sign. The channel factory client credential ~~should~~is recommended to be set to the credential of current application logon user, or from the current user of the logon session. Kerberos authentication protocol is used. For more information about Kerberos authentication protocol, see [RFC4120] and [MS-KILE].

More specifically, implementation ~~should~~is recommended to use WindowsStreamSecurityBindingElement, setting ProtectionLevel = EncryptAndSign.

### 5.1.2   Machine Identity

Channel security using an SSL stream ~~should~~is recommended to be bound to the endpoint, specifying to require client certificate. SSL authentication protocol is used. For more information about SSL authentication protocol, see [RFC2246].

More specifically, use SslStreamSecurityBindingElement, setting RequireClientCertificate to true.<25>

## 5.2   Index of Security Parameters

None.

# 6    Appendix A: 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.

~~Note: Some of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.~~

- Windows 7 operating system with Service Pack 1 (SP1)

- Windows Server 2008 R2 operating system with Service Pack 1 (SP1)

- Windows 8 operating system

- Windows Server 2012 operating system

- ~~Windows 8.1 operating system~~

- Windows Server 2012 R2 operating system

- ~~Windows 10 operating system~~

- Windows Server 2016 ~~Technical Preview~~ operating system

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.

<1> Section 2.2.4:  Windows uses this for non-Windows client platforms.

<2> Section 2.2.4.5.2:  Windows uses this as the Windows SID of this user.

<3> Section 2.2.4.5.2:  Windows uses this as the Windows account of this user.

<4> Section 2.2.4.7.3:  In Windows, this contains non-Windows client binaries.

<5> Section 2.2.4.7.6:  Windows uses this for non-Windows client platforms.

<6> Section 2.2.4.8.2:  The VirtualNetworkSettings type is available in Windows ~~10 and Windows~~ Server 2016 ~~Technical Preview~~ only.

<7> Section 2.2.4.8.3:  The AddressPrefix type is available in Windows ~~10 and Windows~~ Server 2016 ~~Technical Preview~~ only.

<8> Section 2.2.5.6.1:  The implementation is defined by the platform, but could be (as an example) Windows identity.

<9> Section 2.2.5.6.1:  The implementation is defined by the platform, but could be (as an example) **Active Directory** (AD) machine accounts or certificates.

<10> Section 2.2.5.9:  AzureAD and HostedEmail Integration are available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<11> Section 2.2.5.9.2:  InvalidPassword is supported in Windows Server 2012 R2 with [MSKB-2975719] and Windows Server 2016 ~~Technical Preview~~.

<12> Section 3.2.1.2:  In Windows, the schedule can be saved in the Windows task scheduler.

<13> Section 3.4.5.5:  IMachineIdentityProvider.ReturnRemoveMachine is not available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<14> Section 3.4.5.8:  IMachineIdentityProvider.RemoveMachine is not available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<15> Section 3.5.1.1.1.1:  Windows uses this as the Windows account of this user.

<16> Section 3.5.1.1.1.1:  Windows uses this as the Windows SID of this user.

<17> Section 3.5.5.1:  IUserLogonProvider.LogonUser is available only in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<18> Section 3.5.5.2:  IUserChangePasswordProvider.ChangePassword is available only in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<19> Section 3.10.5:  In Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~, the IO365ClientProvider functionality is replaced by IPasswordSyncClientProvider.

<20> Section 3.10.5.2:  The IPasswordSyncClientProvider.RaiseOperationInvokeEvent operation is named Provider.RaiseOperationInvokeEvent in Windows Server 2012.

<21> Section 3.10.5.3:  In Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~, IO365ClientProvider.GetSharePointAddress functionality is replaced by IPasswordSyncClientProvider.

<22> Section 3.10.5.4: IO365ClientProvider.IsActivated is not available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<23> Section 3.10.5.7:  IO365ClientProvider.GetEmailAddress is not available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~.

<24> Section 3.11.1:  Client Notification Service Contract is available in Windows Server 2012 R2 and Windows Server 2016 ~~Technical Preview~~ only.

<25> Section 5.1.2:  In Windows, the certificate can be retrieved using Microsoft.WindowsServerSolutions.Devices.Identity.MachineIdentityManager.GetLocalMachineCert method in the Windows Server operating system Solutions SDK. See [MSDN-GetLocMachCert].

# 7   Change Tracking

No table of This section identifies changes is available. Thethat were made to this document is either new or has had no changes since itsthe last release. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

▪   A document revision that incorporates changes to interoperability requirements or functionality.

▪   The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

▪   New content added.

▪   Content updated.

▪   Content removed.

▪   New product behavior note added.

▪   Product behavior note updated.

▪   Product behavior note removed.

▪   New protocol syntax added.

▪   Protocol syntax updated.

▪   Protocol syntax removed.

▪   New content added due to protocol revision.

▪   Content updated due to protocol revision.

▪   Content removed due to protocol revision.

▪   New protocol syntax added due to protocol revision.

▪   Protocol syntax updated due to protocol revision.

▪   Protocol syntax removed due to protocol revision.

▪   Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|---|---|---|---|
| 2.2.4 Complex Types | 72861 : Added references to the NotificationResult and NotificationResults types. | Y | Content update. |
| 2.2.4.11.1 NotificationResult | 72861 : Added section. | Y | New content added. |
| 2.2.4.11.2 NotificationResults | 72861 : Added section. | Y | New content added. |
| 3.11.5 Message Processing Events and Sequencing Rules | 72861 : Added references to the IClientNotificationProvider.NotifyClientByName, IClientNotificationProvider.NotifyClientById and IClientNotificationProvider.NotifyAllClients operations. | Y | Content update. |
| 3.11.5.3 IClientNotificationProvider.NotifyClientByName | 72861 : Added section. | Y | New content added. |
| 3.11.5.4 IClientNotificationProvider.NotifyClientById | 72861 : Added section. | Y | New content added. |
| 3.11.5.5 IClientNotificationProvider.NotifyAllClients | 72861 : Added section. | Y | New content added. |
| 6 Appendix A: Product Behavior | Updated product applicability list. | Y | Content update. |

# 8 Index

**A**

Applicability 30

**C**

Capability negotiation 30
Change tracking 307
Complex Types message 34

**E**

Elements message 34

**F**

Fields - vendor-extensible 31

**G**

Glossary 27

**I**

Implementer - security considerations 304
Index of security parameters 304
Informative references 29
Introduction 27

**M**

Messages
   Complex Types message 34
   Elements message 34
   Messages message 33
   Namespaces message 32
   Simple Types message 65
   transport 32
Messages message 33

**N**

Namespaces message 32
Normative references 28

**O**

Overview (synopsis) 29

**P**

Parameters - security index 304
Preconditions 30
Prerequisites 30
Product behavior 305

**R**

References 28
   informative 29