[MS-OXWCONFIG]:
Web Service Configuration Protocol Specification

Revision Summary

Date	Revision History	Revision Class	Comments
04/10/2009	.1	Major	Initial Availability.
07/15/2009	1.0	Major	Revised and edited for technical content.
11/04/2009	1.1.0	Minor	Updated the technical content.

Table of Contents

1 Introduction

This document specifies the Web Service Configuration protocol, which sends the request-response messages for retrieving **organization policy** configuration information.

1.1 Glossary

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

Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Socket Layer (HTTPS)
mailbox
Simple Mail Transfer Protocol (SMTP)
SOAP body
SOAP header
Web server
Web Services Description Language (WSDL)
WSDL message
XML
XML schema
XML namespace

The following terms are specific to this document:

organization policy: A policy that is comprised of a condition part and an action part. The condition part is expressed as a sequence of predicates that are evaluated by an e-mail client. The action part specifies what action the e-mail client takes if the condition is met.

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

1.2 References

1.2.1 Normative References

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

[MS-OXGLOS] Microsoft Corporation, "Exchange Server Protocols Master Glossary", June 2008.

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

[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", RFC 3066, January 2001, http://www.ietf.org/rfc/rfc3066.txt.

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

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

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

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

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

1.2.2 Informative References

None.

1.3 Protocol Overview

The Web Service Configuration protocol sends the request-response messages that retrieve organization policy configuration information for a **mailbox**. Clients use the **SOAP** protocol [SOAP1.1] to contact the Web Service Configuration service.

1.4 Relationship to Other Protocols

The Web Service Configuration protocol uses SOAP over **HTTP** and SOAP over **HTTPS**, as shown in the following figures.

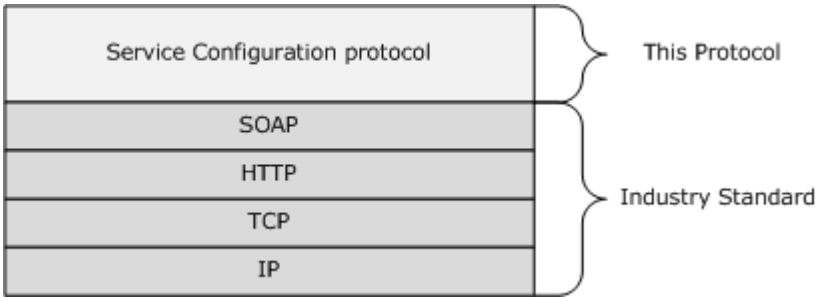

Figure 1: SOAP over HTTP

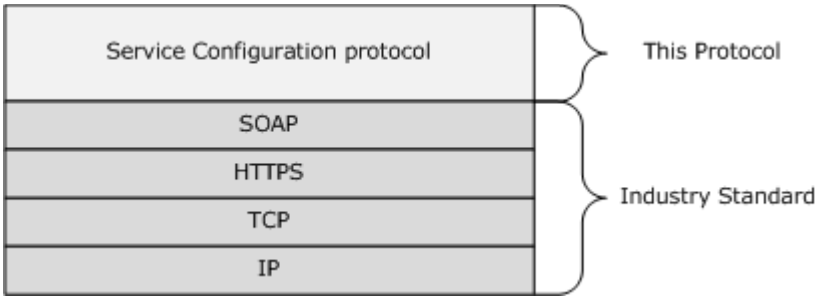

Figure 2: SOAP over HTTPS

1.5 Prerequisites/Preconditions

The URL of the Web Service Configuration protocol can be retrieved by using the Autodiscover Publishing and Lookup SOAP-Based Web Service protocol [MS-OXWSADISC].

1.6 Applicability Statement

The Web Service Configuration protocol can be used when access to organization policy configuration information is available and when communication with the server is enabled for SOAP over HTTP or SOAP over HTTPS.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1, as specified in section 2.1.
- **Protocol Versions:** This protocol specifies only one **WSDL** portType version.
- **Security and Authentication Methods:** This protocol relies on the **Web server** that is hosting it to perform authentication.

Capability Negotiation: None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The SOAP version supported is SOAP 1.1. For details, see [SOAP1.1].

2.2 Common Message Syntax

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

2.2.1 Namespaces

This specification defines and references various **XML namespaces** by using the mechanisms specified in [XMLNS]. 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
soap	http://schemas.xmlsoap.org/WSDL/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWCONFIG]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
targetNamespace	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWCONFIG]
wsdl	http://schemas.xmlsoap.org/WSDL/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	[MS-OXWCONFIG]
m	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWCONFIG]

2.2.2 Simple Types

This specification does not define any common XML schema simple type definitions

2.2.3 Complex Types

This specification does not define any common XML schema complex type definitions.

2.2.4 Elements

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

2.2.5 Attributes

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

2.2.6 Groups

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

2.2.7 Attribute Groups

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

2.2.8 Messages

This specification does not define any common **WSDL message** definitions.

3 Protocol Details

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

3.1 ExchangeServicePortType Server Details

The Web Service Configuration protocol defines a single port type.

```
<wsdl:portType name="ExchangeServicePortType">
   <wsdl:operation name="GetServiceConfiguration">
      <wsdl:input message="tns:GetServiceConfigurationSoapIn" />
      <wsdl:output message="tns:GetServiceConfigurationSoapOut" />
   </wsdl:operation>
</wsdl:portType>
```

3.1.1 Abstract Data Model

The Web Service Configuration protocol is a stateless protocol.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

This protocol includes one operation, **GetServiceConfiguration**, which is stateless and does not have sequencing rules.

Operation	Description
GetServiceConfiguration	The **GetServiceConfiguration** operation provides organization policy configuration information for a mailbox.

3.1.4.1 GetServiceConfiguration

The **GetServiceConfiguration** operation gets the service configuration for a mailbox.

```
<wsdl:operation name="GetServiceConfiguration">
  <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetServiceConfigurat
ion" />
  <wsdl:input>
    <soap:body parts="request" use="literal"/>
    <soap:header message="tns:GetServiceConfigurationSoapIn" part="Impersonation"
use="literal"/>
    <soap:header message="tns:GetServiceConfigurationSoapIn" part="RequestVersion"
use="literal"/>
```

```
    <soap:header message="tns:GetServiceConfigurationSoapIn" part="MailboxCulture"
use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body parts="GetServiceConfigurationResult" use="literal" />
    <soap:header message="tns:GetServiceConfigurationSoapOut" part="ServerVersion"
use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

3.1.4.1.1 Simple Types

The following XML schema simple definitions are specific to the **GetServiceConfiguration** operation.

3.1.4.1.1.1 t:ExchangeVersionType

The **ExchangeVersionType** specifies the server version.

```
<xs:simpleType name="ExchangeVersionType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Exchange2010" />
  </xs:restriction>
</xs:simpleType>
```

Value	Description
Exchange2010	Represents a request against the Exchange2010 schema version.

3.1.4.1.1.2 t:NonEmptyStringType

The **NonEmptyStringType** specifies a string with a minimum length of one character.

```
<xs:simpleType name="NonEmptyStringType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1" />
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.1.3 m:ResponseCodeType

The **ResponseCodeType** specifies the response codes that are returned in a response.

```
<xs:simpleType name="ResponseCodeType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="NoError"/>
        <xs:enumeration value="ErrorAccessDenied"/>
        <xs:enumeration value="ErrorAccountDisabled"/>
        <xs:enumeration value="ErrorADOperation"/>
        <xs:enumeration value="ErrorADSessionFilter"/>
        <xs:enumeration value="ErrorADUnavailable"/>
        <xs:enumeration value="ErrorImpersonateUserDenied"/>
        <xs:enumeration value="ErrorImpersonationDenied"/>
        <xs:enumeration value="ErrorImpersonationFailed"/>
```

```
        <xs:enumeration value="ErrorIncorrectSchemaVersion"/>
        <xs:enumeration value="ErrorInsufficientResources"/>
        <xs:enumeration value="ErrorInternalServerError"/>
        <xs:enumeration value="ErrorInternalServerTransientError"/>
        <xs:enumeration value="ErrorInvalidArgument"/>
        <xs:enumeration value="ErrorInvalidOperation"/>
        <xs:enumeration value="ErrorInvalidRequest"/>
        <xs:enumeration value="ErrorInvalidRoutingType"/>
        <xs:enumeration value="ErrorInvalidServerVersion"/>
        <xs:enumeration value="ErrorMailboxMoveInProgress"/>
        <xs:enumeration value="ErrorMailboxStoreUnavailable"/>
        <xs:enumeration value="ErrorNotEnoughMemory"/>
        <xs:enumeration value="ErrorSchemaValidation"/>
        <xs:enumeration value="ErrorServerBusy"/>
        <xs:enumeration value="ErrorUnsupportedCulture"/>
  <xs:enumeration value="ErrorUnifiedMessagingRequestFailed"/>
    </xs:restriction>
</xs:simpleType>
```

Value	Description
NoError	Specifies that no errors occurred.
ErrorAccessDenied	Specifies that the calling account does not have the rights to perform the requested action.
ErrorAccountDisabled	Specifies that the account in question has been disabled.
ErrorADOperation	Specifies that the **GetServiceConfiguration** operation failed because of communication problems with Active Directory.
ErrorADUnavailable	Specifies that Active Directory is unavailable.
ErrorImpersonateUserDenied	Specifies that the calling account does not have impersonation rights on either the user or contact that it is trying to impersonate or the mailbox database that contains the user mailbox. This response code is returned within a SOAP fault.
ErrorImpersonationDenied	Specifies that the calling account does not have the impersonation right on the server that it is calling. This response code is returned within a SOAP fault.
ErrorImpersonationFailed	Specifies that there was an unexpected error when an attempt was made to perform impersonation. This response code specifies either that the service account that is running the **GetServiceConfiguration** application pool is configured incorrectly, that the **GetServiceConfiguration** service cannot talk to the directory, or that a trust between forests is not correctly configured.
ErrorIncorrectSchemaVersion	Specifies that the request was valid for the current server version but was invalid for the request server version that was specified.
ErrorInsufficientResources	Specifies that the mailbox server is overloaded. Try your request again later.
ErrorInternalServerError	Specifies that the server encountered an error that it could not recover from, and no more specific response code is associated with the error that occurred.

Value	Description
ErrorInternalServerTransientError	Specifies that an internal server error occurred and that you should try your request again later.
ErrorInvalidArgument	Specifies an error related to the **ActingAs** element. This error can occur if the **ActingAs** element is missing, does not include a routing type, does not include an e-mail address, contains an invalid e-mail address, does not resolve to a user in Active Direcotry, or resolves to multiple users in Active Directory.
ErrorInvalidOperation	Specifies a general error that is used when the requested operation is invalid.
ErrorInvalidRequest	Specfies that the SOAP request has a SOAP action header, but nothing in the **SOAP body**.
ErrorInvalidRoutingType	Specifies that the **RoutingType**property that was passed for an **EmailAddressType** is not a valid routing type.
ErrorInvalidServerVersion	Specifies that an invalid request server version was specified in the request.
ErrorMailboxMoveInProgress	Specifies that the mailbox is currently being moved. Try your request again after the move is complete.
ErrorMailboxStoreUnavailable	Specifies that the mailbox database is offline, corrupt, shutting down, or exhibiting other conditions that make the mailbox unavailable.
ErrorNotEnoughMemory	Specifies that the operation could not be completed because of insufficient memory.
ErrorSchemaValidation	Specifies that the incoming request fails schema validation. This response code is always returned within a **SOAP fault**.
ErrorServerBusy	Specifies that the server is busy.
ErrorUnsupportedCulture	Indicates that the value specified for the **Culture** property is not supported.

3.1.4.1.1.4 t:ProtectionRuleActionKindType

The **ProtectionRuleActionKindType** type specifies the actions that are supported by the Protection rules service. Currently, only the *RightsProtectMessage* action is supported. The value MUST be **RightsProtectMessage**.

```
<xs:simpleType name="ProtectionRuleActionKindType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="RightsProtectMessage"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.1.5 t:ProtectionRuleAllInternalType

The **ProtectionRuleAllInternalType** type specifies the AllInternal predicate. The semantics of *AllInternal* is that the predicate matches if all recipients of the e-mail message are internal to the organization of the sender of that e-mail message.

```
<xs:simpleType name="ProtectionRuleAllInternalType">
  <xs:restriction base="xs:string">
    <xs:length value="0"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.1.6 t:ProtectionRuleTrueType

The **ProtectionRuleTrueType** type specifies the True predicate. The semantics of True is that it always matches.

```
<xs:simpleType name="ProtectionRuleTrueType">
  <xs:restriction base="xs:string">
    <xs:length value="0"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.1.7 t:ProtectionRuleValueType

The **ProtectionRuleValueType** type specifies additional arguments to the RecipientIs and SenderDepartments predicates. The value MUST be a string with a minimum length of one character.

```
<xs:simpleType name="ProtectionRuleValueType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.1.8 t:ServiceConfigurationType

The **ServiceConfigurationType** specifies the service configurations that are returned in the response.

```
<xs:simpleType name="ServiceConfigurationType">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="ProtectionRules" />
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

Value	Description
ProtectionRules	Represents the protection rules service configuration.

3.1.4.1.2 Complex Types

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

3.1.4.1.2.1 t: ArrayOfProtectionRulesType

The **ArrayOfProtectionRulesType** type specifies an array of protection rules.

```
<xs:complexType name="ArrayOfProtectionRulesType">
  <xs:sequence>
    <xs:element name="Rule" type="t:ProtectionRuleType" minOccurs="0" maxOccurs="unbounded">
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
Rule	**m:ProtectionRuleType**	Contains a single protection rule. This element can occur zero or more times. This element occurs zero times when no protection rules are defined by the organization. It occurs one or more times if at least one rule is defined by the organization.

3.1.4.1.2.2 m:ArrayOfServiceConfigurationResponseMessageType

The **ArrayOfServiceConfigurationResponseMessageType** type specifies an array of service configuration response messages.

```
<xs:complexType name="ArrayOfServiceConfigurationResponseMessageType">
  <xs:sequence>
    <xs:element name="ServiceConfigurationResponseMessageType"
        type="m:ServiceConfigurationResponseMessageType"
        minOccurs="1" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
ServiceConfigurationResponseMessageType	**m:ServiceConfigurationResponseMessageType**	Contains a service configuration response message. This element MUST occur at least once and can occur two or more times.

3.1.4.1.2.3 m:ArrayOfServiceConfigurationType

The **ArrayOfServiceConfigurationType** type specifies the requested service configurations for a **GetServiceConfigurationSoapIn** message.

```
<xs:complexType name="ArrayOfServiceConfigurationType">
  <xs:choice minOccurs="1" maxOccurs="unbounded">
```

```
    <xs:element name="ConfigurationName" type="t:ServiceConfigurationType"/>
  </xs:choice>
</xs:complexType>
```

Element	Type	Definition
ConfigurationName	**t:ServiceConfigurationType**	Specifies the service configuration that is returned in the response. This element MUST occur at least once.

3.1.4.1.2.4 t:BaseEmailAddressType

The **BaseEmailAddressType** type specifies the base type for the **EmailAddressType** type.

```
<xs:complexType name="BaseEmailAddressType" />
```

3.1.4.1.2.5 m:BaseRequestType

The **BaseRequestType** type is the base for the **GetServiceConfigurationType** type.

```
<xs:complexType name="BaseRequestType" abstract="true"/>
```

3.1.4.1.2.6 t: ConnectingSIDType

The **ConnectingSIDType** type specifies the mailbox identification mechanism for Exchange Impersonation.

```
<xs:complexType name="ConnectingSIDType">
  <xs:choice>
    <xs:element name="PrincipalName" type="t:NonEmptyStringType"/>
    <xs:element name="SID" type="t:NonEmptyStringType"/>
    <xs:element name="PrimarySmtpAddress" type="t:NonEmptyStringType"/>
    <xs:element name="SmtpAddress" type="t:NonEmptyStringType"/>
  </xs:choice>
</xs:complexType>
```

Element	Type	Definition
PrincipalName	**t:NonEmptyStringType**	Specifies the mailbox by user principal name. The **SID**, **PrimarySmtpAddress**, and **SmtpAddress** elements cannot be used if the **PrincipalName** element is used.
SID	**t:NonEmptyStringType**	Specifies the mailbox by security identifier. The **PrincipalName**, **PrimarySmtpAddress**, and **SmtpAddress** elements cannot be used if the **SID** element is used.
PrimarySmtpAddress	**t:NonEmptyStringType**	Specifies the mailbox by primary **SMTP** address. The **SID**, **PrincipalName**, and **SmtpAddress** elements cannot be used if the **PrimarySmtpAddress** element is used.

Element	Type	Definition
SmtpAddress	**t:NonEmptyStringType**	Specifies the mailbox by SMTP address. The **SID**, **PrimarySmtpAddress**, and **PrincipalName** elements cannot be used if the **SmtpAddress** element is used.

3.1.4.1.2.7 t:EmailAddressType

The **EmailAddressType** type specifies the identity of the sender or recipient actor of the request. The **EmailAddressType** type extends the **BaseEmailAddressType**.

```
<xs:complexType name="EmailAddressType">
  <xs:complexContent>
    <xs:extension base="t:BaseEmailAddressType">
      <xs:sequence>
        <xs:element name="EmailAddress" type="t:NonEmptyStringType" minOccurs="0"/>
        <xs:element name="RoutingType" type="t:NonEmptyStringType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Element	Type	Definition
EmailAddress	**NonEmptyStringType**	Specifies the SMTP address of the requested user's configuration information.
RoutingType	**NonEmptyStringType**	Specifies the routing type for the e-mail address.

3.1.4.1.2.8 t: ExchangeImpersonationType

The **ExchangeImpersonationType** type specifies the account to impersonate by using Exchange Impersonation.

```
<xs:complexType name="ExchangeImpersonationType">
  <xs:sequence>
    <xs:element name="ConnectingSID" type="t:ConnectingSIDType" />
  </xs:sequence>
  <xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
  </xs:anyAttribute>
</xs:complexType>
```

Element	Type	Definition
ConnectingSID	**t:ConnectingSIDType**	Specifies the mailbox identification mechanism for Exchange Impersonation. This element MUST occur once if Exchange Impersonation is used.

3.1.4.1.2.9 m:GetServiceConfigurationResponseMessageType

The **GetServiceConfigurationResponseMessageType** type contains the response message for a **GetServiceConfiguration** operation. The **GetServiceConfigurationResponseMessageType** type extends the **ResponseMessageType**.

```
<xs:complexType name="GetServiceConfigurationResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="ResponseMessages"
type="m:ArrayOfServiceConfigurationResponseMessageType" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Element	Type	Definition
ResponseMessages	**m:ArrayOfServiceConfigurationResponseMessageType**	Contains an array of service configuration responsemessages. This element MUST occur if there are configuration settings.

3.1.4.1.2.10 m:GetServiceConfigurationType

The **GetServiceConfigurationType** type specifies the requested service configurations and identifies the sender or recipient actor who is making the request. The recipient actor can be different from the user who authenticated with the server. The **GetServiceConfigurationType** type extends the **BaseRequestType** type.

```
<xs:complexType name="GetServiceConfigurationType">
  <xs:complexContent>
    <xs:extension base="m:BaseRequestType">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="ActingAs" type="t:EmailAddressType"/>
        <xs:element minOccurs="1" maxOccurs="1" name="RequestedConfiguration"
type="m:ArrayOfServiceConfigurationType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Element	Type	Definition
ActingAs	**t:EmailAddressType**	Specifies who the caller is sending as. If this element is not present, the authenticated user is assumed to be the sender. The **ActingAs** element MUST be included for requesting sender hints. This element is optional.
RequestedConfiguration	**m:ArrayOfServiceConfigurationType**	Specifies the requested service configurations. This element MUST be present.

3.1.4.1.2.11 t:MailboxCultureType

The **MailboxCultureType** type specifies the mailbox language. Values are defined in [RFC3066]. This is used in the **SOAP header**. This type extends the **xs:language** type specified in [XMLSCHEMA2].

```
<xs:complexType name="MailboxCultureType">
  <xs:simpleContent>
    <xs:extension base="xs:language">
      <xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
      </xs:anyAttribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

3.1.4.1.2.12 t:ProtectionRuleActionType

The **ProtectionRuleActionType** type specifies the action that the client can take if the condition part of the associated rule matches.

```
<xs:complexType name="ProtectionRuleActionType">
  <xs:sequence>
    <xs:element name="Argument" type="t:ProtectionRuleArgumentType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="Name" use="required" type="t:ProtectionRuleActionKindType" />
</xs:complexType>
```

Element	Type	Definition
Argument	**t:ProtectionRuleArgumentType**	Specifies arguments to the action. This element MUST NOT occur if the specified action does not require arguments to be specified. This element can occur one or more times if an action requires one or more arguments. The **RightsProtectMessage**action MUST contain a single argument.

Attribute	Type	Definition
Name	**t:ProtectionRuleActionKindType**	Specifies the name of the action. This attribute MUST be present.

3.1.4.1.2.13 t:ProtectionRuleAndType

The **ProtectionRuleAndType** type specifies that there MUST be more than one protection rule condition. Elements of type **ProtectionRuleAndType** MUST contain at least one child element.

```
<xs:complexType name="ProtectionRuleAndType">
  <xs:sequence>
    <xs:choice minOccurs="1" maxOccurs="unbounded">
      <xs:element name="AllInternal" type="t:ProtectionRuleAllInternalType" />
      <xs:element name="And" type="t:ProtectionRuleAndType" />
```

```
      <xs:element name="RecipientIs" type="t:ProtectionRuleRecipientIsType" />
      <xs:element name="SenderDepartments" type="t:ProtectionRuleSenderDepartmentsType" />
      <xs:element name="True" type="t:ProtectionRuleTrueType" />
    </xs:choice>
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
AllInternal	**t:ProtectionRuleAllInternalType**	Evaluates to **TRUE** if all recipients of an e-mail message are internal to the sender's organization.
And	**t:ProtectionRuleAndType**	Specifies that all child elements MUST match to evaluate to **TRUE**.
RecipientIs	**t:ProtectionRuleRecipientIsType**	The **RecipientIs**condition evaluates to **TRUE** if any recipient of the e-mail Message matches any of the specified recipients in the child **Value** elements.
SenderDepartments	**t:ProtectionRuleSenderDepartmentsType**	The **SenderDepartments**condition evaluates to **TRUE** if the department of the sender matches any specified department in the child **Value** elements.
True	**t:ProtectionRuleTrueType**	Specifies a condition that always matches.

3.1.4.1.2.14 t:ProtectionRuleArgumentType

The **ProtectionRuleArgumentType** type specifies an attribute that is used to specify an argument to an action. The protection is identified by the **Value** attribute.

```
<xs:complexType name="ProtectionRuleArgumentType">
  <xs:attribute name="Value" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
```

Attribute	Type	Definition
Value	**xs:string**	Specifies the value of an argument to the action part of a protection rule. This attribute value MUST be a string of at least one character in length.

3.1.4.1.2.15 t:ProtectionRuleConditionType

The **ProtectionRuleConditionType** type specifies the condition part of a protection rule.

```
<xs:complexType name="ProtectionRuleConditionType">
  <xs:choice minOccurs="1" maxOccurs="1">
    <xs:element name="AllInternal" type="t:ProtectionRuleAllInternalType" />
    <xs:element name="And" type="t:ProtectionRuleAndType" />
    <xs:element name="RecipientIs" type="t:ProtectionRuleRecipientIsType" />
    <xs:element name="SenderDepartments" type="t:ProtectionRuleSenderDepartmentsType" />
    <xs:element name="True" type="t:ProtectionRuleTrueType" />
  </xs:choice>
</xs:complexType>
```

Element	Type	Definition
AllInternal	**t:ProtectionRuleAllInternalType**	The **AllInternal** condition evaluates to **TRUE** if all recipients of an e-mail message are internal to the sender's organization. If this element exists, the **And**, **RecipientsIs**, **SenderDeprtments**, and **True** elements MUST NOT exist as a direct child node of elements of type **ProtectionRuleConditionType**.
And	**t:ProtectionRuleAndType**	Specifies that all child elements MUST match to evaluate to **TRUE**. Specifies that there MUST be more than one protection rule child condition. If this element exists, the **AllInternal**, **RecipientsIs**, **SenderDeprtments**, and **True** elements MUST NOT exist as a direct child node of elements of type **ProtectionRuleConditionType**.
RecipientIs	**t:ProtectionRuleRecipientIsType**	Specifies that any recipient of the e-mail message matches any of the specified recipients in the child **Value** elements. If this element exists, the **And**, **AllInternal**, **SenderDeprtments**, and **True** elements MUST NOT exist as a direct child node of elements of type **ProtectionRuleConditionType**.
SenderDepartments	**t:ProtectionRuleSenderDepartmentsType**	Specifies that the department of the sender matches any of the

Element	Type	Definition
		specified departments in the child **Value** elements. If this element exists, the **And**, **RecipientsIs**, **AllInternal**, and **True** elements MUST NOT exist as a direct child node of elements of type **ProtectionRuleConditionType**.
True	**t:ProtectionRuleTrueType**	Specifies a condition that always matches. If this element exists, the **And**, **RecipientsIs**, **SenderDeprtments**, and **AllInternal** elements MUST NOT exist as a direct child node of elements of type **ProtectionRuleConditionType**.

3.1.4.1.2.16 t:ProtectionRuleRecipientIsType

The **ProtectionRuleRecipientIsType** type specifies a condition that matches if any recipients of the e-mail message match any specified recipients in the child **Value** elements.

```
<xs:complexType name="ProtectionRuleRecipientIsType">
  <xs:sequence>
    <xs:element name="Value" type="t:ProtectionRuleValueType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
Value	**t: ProtectionRuleValueType**	Specifies an argument to the **RecipientIs** condition. This element MUST occur at least once.

3.1.4.1.2.17 t:ProtectionRuleSenderDepartmentsType

The **ProtectionRuleSenderDepartmentsType** type specifies a condition that matches if the department of the sender of the e-mail message matches any of the specified departments in the child **Value** elements.

```
<xs:complexType name="ProtectionRuleSenderDepartmentsType">
  <xs:sequence>
    <xs:element name="Value" type="t:ProtectionRuleValueType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
Value	**t: ProtectionRuleValueType**	Specifies an argument to the **SenderDepartments** condition.

Element	Type	Definition
		This element MUST occur at least once.

3.1.4.1.2.18 t:ProtectionRulesServiceConfiguration

The **ProtectionRulesServiceConfiguration** type specifies the configuration of the protection rules service. The configuration is comprised of a list of rules, internal domains, and a refresh interval. The **ProtectionRulesServiceConfiguration** type extends the **ServiceConfiguration** type.

```
<xs:complexType name="ProtectionRulesServiceConfiguration">
  <xs:complexContent>
    <xs:extension base="t:ServiceConfiguration">
      <xs:sequence>
        <xs:element name="Rules" type="t:ArrayOfProtectionRulesType" minOccurs="1"
maxOccurs="1" />
        <xs:element name="InternalDomains" type="t:SmtpDomainList" minOccurs="1"
maxOccurs="1" />
      </xs:sequence>
      <xs:attribute name="RefreshInterval" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:int">
            <xs:minInclusive value="1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Element	Type	Definition
Rules	**t:ArrayOfProtectionRulesType**	Specifies the collection of rules to be evaluated. This element MUST be included if the **ProtectionRulesServiceConfiguration** type is used.
InternalDomains	**t:SmtpDomainList**	Specifies the list of internal SMTPdomains of the organization. This element MUST be included if the **ProtectionRulesServiceConfiguration** type is used.

Attribute	Type	Definition
RefreshInterval	**xs:int**	Specifies how often, in whole hours, the client SHOULD request protection rules from the server. This element MUST be included if the **ProtectionRulesServiceConfiguration** type is used.

3.1.4.1.2.19 t:ProtectionRuleType

The **ProtectionRuleType** type specifies a single protection rule.

```
<xs:complexType name="ProtectionRuleType">
  <xs:sequence>
    <xs:element name="Condition" type="t:ProtectionRuleConditionType" minOccurs="1"
maxOccurs="1"/>
    <xs:element name="Action" type="t:ProtectionRuleActionType" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="Name" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="UserOverridable" type="xs:boolean" use="required" />
  <xs:attribute name="Priority" use="required" >
    <xs:simpleType>
      <xs:restriction base="xs:int">
        <xs:minInclusive value="1"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
```

Element	Type	Definition
Condition	**t:ProtectionRuleConditionType**	Specifies the condition that MUST be satisfied for the action part of the rule to be executed. This element MUST be included if the **ProtectionRuleType** type is used.
Action	**t:ProtectionRuleActionType**	Specifies what action MUST be executed if the condition part of the rule matches. This element MUST be included if the **ProtectionRuleType** type is used.

Attribute	Type	Definition
Name	**xs:string**	Specifies the name of the rule. This attribute MUST be included if the **ProtectionRuleType** type is used. This attribute value MUST contain a string of at least one character.
UserOverridable	**xs:boolean**	Specifies whether the rule is mandatory. If the rule is mandatory, this attribute value MUST be **FALSE**. This attribute MUST be included if the **ProtectionRuleType** type is used.
Priority	**xs:int**	Specifies the rule priority. The lower bound MUST be 1. This attribute MUST be included if the **ProtectionRuleType** type is used. This attribute value MUST contain an integer value of at least one.

3.1.4.1.2.20 m:ResponseMessageType

The **ResponseMessageType** type specifies the base type for the **GetServiceConfigurationResponseMessageType** type.

```
<xs:complexType name="ResponseMessageType">
    <xs:sequence minOccurs="0">
        <xs:element name="MessageText" type="xs:string" minOccurs="0"/>
        <xs:element name="ResponseCode" type="m:ResponseCodeType" minOccurs="0"/>
        <xs:element name="DescriptiveLinkKey" type="xs:int" minOccurs="0"/>
        <xs:element name="MessageXml" minOccurs="0">
            <xs:complexType>
                <xs:sequence>
                    <xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:complexType>

        </xs:element>
    </xs:sequence>
    <xs:attribute name="ResponseClass" type="t:ResponseClassType" use="required"/>
</xs:complexType>
```

Element	Type	Definition
MessageText	**xs:string**	Specifies the status of the response. The following values are valid for this attribute: Success Warning Error
ResponseCode	**m:ResponseCodeType**	Specifies an error code that identifies the specific error that the request encountered. Can be present.
DescriptiveLinkKey	**xs:int**	Currently unused, and is reserved for future use. It contains a value of 0 (zero).
MessageXml	**Complex type defined above.**	Provides additional error response information. Can be present.

3.1.4.1.2.21 t:ServiceConfiguration

The **ServiceConfiguration** type specifies the base type for the service configuration types.

```
<xs:complexType name="ServiceConfiguration">
</xs:complexType>
```

3.1.4.1.2.22 m:ServiceConfigurationResponseMessageType

The **ServiceConfigurationResponseMessageType** type specifies service configuration settings. The **ServiceConfigurationResponseMessageType** type extends the **ResponseMessageType**.

```
<xs:complexType name="ServiceConfigurationResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
```

```
      <xs:sequence>
        <xs:element name="ProtectionRulesConfiguration"
type="t:ProtectionRulesServiceConfiguration" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Element	Type	Definition
ProtectionRulesConfiguration	**t:ProtectionRulesServiceConfiguration**	Contains service configuration for the protection rules service. This element **MUST** occur for protection rules configuration.

3.1.4.1.2.23 t:SmtpDomain

The **SmtpDomain** type specifies a single domain.

```
<xs:complexType name="SmtpDomain">
  <xs:attribute name="Name" type="xs:string" use="required"/>
  <xs:attribute name="IncludeSubdomains" type="xs:boolean" use="optional"/>
</xs:complexType>
```

Attribute	Type	Definition
Name	**xs:string**	Specifies the name of a domain. This attribute MUST be set.
IncludeSubdomains	**xs:boolean**	Specifies whether sub-domains of the domain identified by the **Name** attribute are considered internal.

3.1.4.1.2.24 t:SmtpDomainList

The **SmtpDomainList** type specifies a list of internal domains in a user's organization.

```
<xs:complexType name="SmtpDomainList">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Domain" type="t:SmtpDomain"/>
  </xs:sequence>
</xs:complexType>
```

Element	Type	Definition
Domain	**t:SmtpDomain**	Specifies a single SMTPdomain. This element can occur 0 or more times.

3.1.4.1.3 Elements

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

3.1.4.1.3.1 t:ExchangeImpersonation

The **ExchangeImpersonation** element specifies the mailbox to impersonate.

```
<xs:element name="ExchangeImpersonation" type="t:ExchangeImpersonationType" />
```

3.1.4.1.3.2 m:GetServiceConfiguration

The **GetServiceConfiguration** element specifies the base element for a **GetServiceConfiguration** request.

```
<xs:element name="GetServiceConfiguration" type="m:GetServiceConfigurationType"/>
```

3.1.4.1.3.3 m:GetServiceConfigurationResponse

The **GetServiceConfigurationResponse** element specifies the response message for a **GetServiceConfiguration** operation.

```
<xs:element name="GetServiceConfigurationResponse"
type="m:GetServiceConfigurationResponseMessageType"/>
```

3.1.4.1.3.4 t:MailboxCulture

The **MailboxCulture** element specifies the mailbox language.

```
<xs:element name="MailboxCulture" type="t:MailboxCultureType"/>
```

3.1.4.1.3.5 t:ServerVersionInfo

The **ServerVersionInfo** element specifies the server version. This is returned in the **GetServiceConfigurationSoapOut** message.

```
<xs:element name="ServerVersionInfo">
  <xs:complexType>
    <xs:attribute name="MajorVersion" type="xs:int" use="optional"/>
    <xs:attribute name="MinorVersion" type="xs:int" use="optional"/>
    <xs:attribute name="MajorBuildNumber" type="xs:int" use="optional"/>
    <xs:attribute name="MinorBuildNumber" type="xs:int" use="optional"/>
    <xs:attribute name="Version" type="xs:string" use="optional"/>
  </xs:complexType>
</xs:element>
```

Attribute	Type	Definition
MajorVersion	**xs:int**	Specifies the major server version. This attribute MUST be present in all responses.
MinorVersion	**xs:int**	Specifies the minor server version. This attribute MUST be present in all responses.
MajorBuildNumber	**xs:int**	Specifies the major server build number. This attribute MUST be

Attribute	Type	Definition
		present in all responses.
MinorBuildNumber	**xs:int**	Specifies the minor server build number. This attribute MUST be present in all responses.
Version	**xs:string**	Specifies the server version. This attribute MUST be present.

3.1.4.1.3.6 t:RequestServerVersion

The **RequestServerVersion** element specifies the server version to target for the **GetServiceConfiguration** operation.

```
<xs:element name="RequestServerVersion">
  <xs:complexType>
    <xs:attribute name="Version" type="t:ExchangeVersionType" fixed="Exchange2010"
use="required"/>
    <xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
    </xs:anyAttribute>
  </xs:complexType>
</xs:element>
```

Attribute	Type	Definition
Version	**t:ExchangeVersionType**	Specifies the server version to target. This attribute MUST be present in all **GetServiceConfigurationSoapIn** messages.

3.1.4.1.4 Attributes

This specification does not define any specific XML schema attribute definitions for this operation.

3.1.4.1.5 Groups

This specification does not define any specific XML schema group definitions for this operation.

3.1.4.1.6 Attribute Groups

This specification does not define any specific XML schema attribute group definitions for this operation.

3.1.4.1.7 Messages

The following WSDL message definitions are specific to this operation.

3.1.4.1.7.1 GetServiceConfigurationSoapIn

The **GetServiceConfigurationSoapIn** message contains four parts.

Part Name	Element/Type	Description
request	**GetServiceConfiguration**	This part specifies the request.
Impersonation	**ExchangeImpersonation**	This part specifies Exchange Impersonation information.

Part Name	Element/Type	Description
RequestVersion	**RequestServerVersion**	This part specifies the schema version for the **GetServiceConfiguration** request.
MailboxCulture	**MailboxCulture**	This part specifies the culture to use for accessing the mailbox. The cultures are defined by [RFC3066].

3.1.4.1.7.2 GetServiceConfigurationSoapOut

The **GetServiceConfigurationSoapOut** message contains two parts.

Part Name	Element/Type	Description
GetServiceConfigurationResult	**GetServiceConfigurationResponse**	This part specifies the response.
ServerVersion	**ServerVersionInfo**	This part specifies the server version for the response.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

3.2 Client Details

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

3.2.1 Abstract Data Model

None.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Message Processing Events and Sequencing Rules

None.

3.2.5 Timer Events

None.

3.2.6 Other Local Events

None.

4 Protocol Examples

4.1 GetServiceConfiguration Request

The following example shows how to get sender hints for a user.

```
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xmlns:xsd="http://www.w3.org/2001/XMLSchema"
               xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
               xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
  <soap:Header>
    <t:RequestServerVersion Version="Exchange2010" />
  </soap:Header>

  <soap:Body>
    <GetServiceConfiguration
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <ActingAs>
        <t:EmailAddress>user1@contoso.com</t:EmailAddress>
        <t:RoutingType>SMTP</t:RoutingType>
      </ActingAs>
      <RequestedConfiguration>
        <ConfigurationName>MailTips</ConfigurationName>
      </RequestedConfiguration>
    </GetServiceConfiguration>
  </soap:Body>

</soap:Envelope>
```

4.2 GetServiceConfiguration Response

The following is an example of a successful response from the **GetServiceConfiguration** service.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
                         MinorVersion="0"
                         MajorBuildNumber="482"
                         MinorBuildNumber="17"
                         Version="Exchange2010"
                         xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
                         xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
                         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                         xmlns:xsd="http://www.w3.org/2001/XMLSchema"/>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetServiceConfigurationResponse ResponseClass="Success"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <ResponseCode>NoError</ResponseCode>
      <ResponseMessages>
        <ServiceConfigurationResponseMessageType ResponseClass="Success">
          <ResponseCode>NoError</ResponseCode>
          <m:MailTipsConfiguration
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
```

```
            <t:MaxRecipientsPerGetMailTipsRequest
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">50</t:MaxRecipientsPerGet
MailTipsRequest>
            <t:MaxMessageSize
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">10485760</t:MaxMessageSiz
e>
            <t:LargeAudienceThreshold
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">25</t:LargeAudienceThresh
old>
            <t:ShowExternalRecipientCount
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">false</t:ShowExternalReci
pientCount>
            <t:InternalDomains
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
              <t:Domain Name="contoso.com" IncludeSubdomains="false"/>
              <t:Domain Name="fabrikam.com" IncludeSubdomains="false"/>
              <t:Domain Name="example.com" IncludeSubdomains="false"/>
            </t:InternalDomains>
          </m:MailTipsConfiguration>
        </ServiceConfigurationResponseMessageType>
      </ResponseMessages>
    </GetServiceConfigurationResponse>
  </s:Body>
</s:Envelope>
```

4.3 Unsuccessful Response

4.3.1 SOAP Exception

The following is an example of a SOAP fault caused by the request failing schema validation.

```
<?xml version="1.0" encoding="utf-8"?><s:Envelope
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <s:Fault>
      <faultcode
xmlns:a="http://schemas.microsoft.com/exchange/services/2006/types">a:ErrorSchemaValidation</
faultcode>
      <faultstring xml:lang="en-US">The request failed schema validation: The element
'RequestedConfiguration' in namespace
'http://schemas.microsoft.com/exchange/services/2006/messages' has incomplete content. List
of possible elements expected: 'ConfigurationName' in namespace
'http://schemas.microsoft.com/exchange/services/2006/messages'.</faultstring>
      <detail>
        <e:ResponseCode
xmlns:e="http://schemas.microsoft.com/exchange/services/2006/errors">ErrorSchemaValidation</e
:ResponseCode>
        <e:Message xmlns:e="http://schemas.microsoft.com/exchange/services/2006/errors">The
request failed schema validation.</e:Message>
        <e:MessageXml xmlns:e="http://schemas.microsoft.com/exchange/services/2006/errors">
          <t:LineNumber
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">16</t:LineNumber>
          <t:LinePosition
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">9</t:LinePosition>
          <t:Violation
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">The element
'RequestedConfiguration' in namespace
'http://schemas.microsoft.com/exchange/services/2006/messages' has incomplete content. List
```

```
of possible elements expected: 'ConfigurationName' in namespace
'http://schemas.microsoft.com/exchange/services/2006/messages'.</t:Violation>
        </e:MessageXml>
      </detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
```

4.3.2 GetServiceConfiguration Error Response

The following is an example of a user specified in the **ActingAs** element not being found in the directory.

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:ServerVersionInfo MajorVersion="14"
                         MinorVersion="0"
                         MajorBuildNumber="482"
                         MinorBuildNumber="17"
                         Version="Exchange2010"
                         xmlns:h="http://schemas.microsoft.com/exchange/services/2006/types"
                         xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
                         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                         xmlns:xsd="http://www.w3.org/2001/XMLSchema"/>
  </s:Header>
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetServiceConfigurationResponse ResponseClass="Error"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <MessageText>The ActingAs parameter does not match a user in the
directory.</MessageText>
      <ResponseCode>ErrorInvalidArgument</ResponseCode>
      <DescriptiveLinkKey>0</DescriptiveLinkKey>
    </GetServiceConfigurationResponse>
  </s:Body>
</s:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

The Service Configuration service does not use additional security mechanisms.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

See [WSDL] for a specification of Web Services Descriptor Language (WSDL).

For ease of implementation, the full WSDL is provided.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:s="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
<wsdl:types>
<xs:schema id="messages" elementFormDefault="qualified" version="Exchange2010"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
<xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"/>
<xs:simpleType name="ResponseCodeType">
<xs:restriction base="xs:string">
<xs:enumeration value="NoError"/>
<xs:enumeration value="ErrorAccessDenied"/>
<xs:enumeration value="ErrorAccountDisabled"/>
<xs:enumeration value="ErrorADOperation"/>
<xs:enumeration value="ErrorADUnavailable"/>
<xs:enumeration value="ErrorImpersonateUserDenied"/>
<xs:enumeration value="ErrorImpersonationDenied"/>
<xs:enumeration value="ErrorImpersonationFailed"/>
<xs:enumeration value="ErrorIncorrectSchemaVersion"/>
<xs:enumeration value="ErrorInsufficientResources"/>
<xs:enumeration value="ErrorInternalServerError"/>
<xs:enumeration value="ErrorInternalServerTransientError"/>
<xs:enumeration value="ErrorInvalidArgument"/>
<xs:enumeration value="ErrorInvalidOperation"/>
<xs:enumeration value="ErrorInvalidRequest"/>
<xs:enumeration value="ErrorInvalidRoutingType"/>
<xs:enumeration value="ErrorInvalidServerVersion"/>
<xs:enumeration value="ErrorMailboxMoveInProgress"/>
<xs:enumeration value="ErrorMailboxStoreUnavailable"/>
<xs:enumeration value="ErrorNotEnoughMemory"/>
<xs:enumeration value="ErrorSchemaValidation"/>
<xs:enumeration value="ErrorServerBusy"/>
<xs:enumeration value="ErrorUnsupportedCulture"/>
</xs:restriction>
</xs:simpleType>
<xs:complexType name="ArrayOfServiceConfigurationResponseMessageType">
<xs:sequence>
<xs:element name="ServiceConfigurationResponseMessageType"
type="m:ServiceConfigurationResponseMessageType" minOccurs="1" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfServiceConfigurationType">
<xs:choice minOccurs="1" maxOccurs="unbounded">
<xs:element name="ConfigurationName" type="t:ServiceConfigurationType"/>
</xs:choice>
```

```
</xs:complexType>
<xs:complexType name="BaseRequestType" abstract="true"/>
<xs:complexType name="GetServiceConfigurationResponseMessageType">
<xs:complexContent>
<xs:extension base="m:ResponseMessageType">
<xs:sequence>
<xs:element name="ResponseMessages" type="m:ArrayOfServiceConfigurationResponseMessageType"
minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="GetServiceConfigurationType">
<xs:complexContent>
<xs:extension base="m:BaseRequestType">
<xs:sequence>
<xs:element minOccurs="0" maxOccurs="1" name="ActingAs" type="t:EmailAddressType"/>
<xs:element minOccurs="1" maxOccurs="1" name="RequestedConfiguration"
type="m:ArrayOfServiceConfigurationType"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ResponseMessageType">
<xs:sequence minOccurs="0">
<xs:element name="MessageText" type="xs:string" minOccurs="0"/>
<xs:element name="ResponseCode" type="m:ResponseCodeType" minOccurs="0"/>
<xs:element name="DescriptiveLinkKey" type="xs:int" minOccurs="0"/>
<xs:element name="MessageXml" minOccurs="0">
<xs:complexType>
<xs:sequence>
<xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="ResponseClass" type="t:ResponseClassType" use="required"/>
</xs:complexType>
<xs:complexType name="ServiceConfigurationResponseMessageType">
<xs:complexContent>
<xs:extension base="m:ResponseMessageType">
<xs:sequence>
<xs:element name="ProtectionRulesConfiguration" type="t:ProtectionRulesServiceConfiguration"
minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:element name="GetServiceConfiguration" type="m:GetServiceConfigurationType"/>
<xs:element name="GetServiceConfigurationResponse"
type="m:GetServiceConfigurationResponseMessageType"/>
</xs:schema>
<xs:schema id="types" elementFormDefault="qualified" version="Exchange2010"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
<xs:simpleType name="ResponseClassType">
```

```
<xs:restriction base="xs:string">
<xs:enumeration value="Success"/>
<xs:enumeration value="Warning"/>
<xs:enumeration value="Error"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ExchangeVersionType">
<xs:restriction base="xs:string">
<xs:enumeration value="Exchange2010"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="NonEmptyStringType">
<xs:restriction base="xs:string">
<xs:minLength value="1"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProtectionRuleActionKindType">
<xs:restriction base="xs:string">
<xs:enumeration value="RightsProtectMessage"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProtectionRuleAllInternalType">
<xs:restriction base="xs:string">
<xs:length value="0"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProtectionRuleTrueType">
<xs:restriction base="xs:string">
<xs:length value="0"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProtectionRuleValueType">
<xs:restriction base="xs:string">
<xs:minLength value="1"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="ServiceConfigurationType">
<xs:list>
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="ProtectionRules"/>
</xs:restriction>
</xs:simpleType>
</xs:list>
</xs:simpleType>
<xs:complexType name="ArrayOfProtectionRulesType">
<xs:sequence>
<xs:element name="Rule" type="t:ProtectionRuleType" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="BaseEmailAddressType"/>
<xs:complexType name="BaseItemIdType" abstract="true"/>
<xs:complexType name="ConnectingSIDType">
<xs:choice>
<xs:element name="PrincipalName" type="t:NonEmptyStringType"/>
<xs:element name="SID" type="t:NonEmptyStringType"/>
<xs:element name="PrimarySmtpAddress" type="t:NonEmptyStringType"/>
<xs:element name="SmtpAddress" type="t:NonEmptyStringType"/>
</xs:choice>
```

```
</xs:complexType>
<xs:complexType name="EmailAddressType">
<xs:complexContent>
<xs:extension base="t:BaseEmailAddressType">
<xs:sequence>
<xs:element name="EmailAddress" type="t:NonEmptyStringType" minOccurs="0"/>
<xs:element name="RoutingType" type="t:NonEmptyStringType" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ExchangeImpersonationType">
<xs:sequence>
<xs:element name="ConnectingSID" type="t:ConnectingSIDType"/>
</xs:sequence>
<xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
        </xs:anyAttribute>
</xs:complexType>

<xs:complexType name="MailboxCultureType">
<xs:simpleContent>
<xs:extension base="xs:language">
<xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
            </xs:anyAttribute>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

<xs:complexType name="ProtectionRuleActionType">
<xs:sequence>
<xs:element name="Argument" type="t:ProtectionRuleArgumentType" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="Name" use="required" type="t:ProtectionRuleActionKindType"/>
</xs:complexType>
<xs:complexType name="ProtectionRuleAndType">
<xs:sequence>
<xs:choice minOccurs="1" maxOccurs="unbounded">
<xs:element name="AllInternal" type="t:ProtectionRuleAllInternalType"/>
<xs:element name="And" type="t:ProtectionRuleAndType"/>
<xs:element name="RecipientIs" type="t:ProtectionRuleRecipientIsType"/>
<xs:element name="SenderDepartments" type="t:ProtectionRuleSenderDepartmentsType"/>
<xs:element name="True" type="t:ProtectionRuleTrueType"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ProtectionRuleArgumentType">
<xs:attribute name="Value" use="required">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:minLength value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>
<xs:complexType name="ProtectionRuleConditionType">
<xs:choice minOccurs="1" maxOccurs="1">
<xs:element name="AllInternal" type="t:ProtectionRuleAllInternalType"/>
<xs:element name="And" type="t:ProtectionRuleAndType"/>
```

```
<xs:element name="RecipientIs" type="t:ProtectionRuleRecipientIsType"/>
<xs:element name="SenderDepartments" type="t:ProtectionRuleSenderDepartmentsType"/>
<xs:element name="True" type="t:ProtectionRuleTrueType"/>
</xs:choice>
</xs:complexType>
<xs:complexType name="ProtectionRuleRecipientIsType">
<xs:sequence>
<xs:element name="Value" type="t:ProtectionRuleValueType" minOccurs="1"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ProtectionRuleSenderDepartmentsType">
<xs:sequence>
<xs:element name="Value" type="t:ProtectionRuleValueType" minOccurs="1"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ProtectionRulesServiceConfiguration">
<xs:complexContent>
<xs:extension base="t:ServiceConfiguration">
<xs:sequence>
<xs:element name="Rules" type="t:ArrayOfProtectionRulesType" minOccurs="1" maxOccurs="1"/>
<xs:element name="InternalDomains" type="t:SmtpDomainList" minOccurs="1" maxOccurs="1"/>
</xs:sequence>
<xs:attribute name="RefreshInterval" use="required">
<xs:simpleType>
<xs:restriction base="xs:int">
<xs:minInclusive value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="ProtectionRuleType">
<xs:sequence>
<xs:element name="Condition" type="t:ProtectionRuleConditionType" minOccurs="1"
maxOccurs="1"/>
<xs:element name="Action" type="t:ProtectionRuleActionType" minOccurs="1" maxOccurs="1"/>
</xs:sequence>
<xs:attribute name="Name" use="required">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:minLength value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="UserOverridable" type="xs:boolean" use="required"/>
<xs:attribute name="Priority" use="required">
<xs:simpleType>
<xs:restriction base="xs:int">
<xs:minInclusive value="1"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>
<xs:complexType name="ServiceConfiguration"/>
<xs:complexType name="SmtpDomain">
<xs:attribute name="Name" type="xs:string" use="required"/>
```

```
<xs:attribute name="IncludeSubdomains" type="xs:boolean" use="optional"/>
</xs:complexType>
<xs:complexType name="SmtpDomainList">
<xs:sequence>
<xs:element minOccurs="0" maxOccurs="unbounded" name="Domain" type="t:SmtpDomain"/>
</xs:sequence>
</xs:complexType>

<xs:element name="ExchangeImpersonation" type="t:ExchangeImpersonationType"/>
<xs:element name="MailboxCulture" type="t:MailboxCultureType"/>
<xs:element name="ServerVersionInfo">
<xs:complexType>
<xs:attribute name="MajorVersion" type="xs:int" use="optional"/>
<xs:attribute name="MinorVersion" type="xs:int" use="optional"/>
<xs:attribute name="MajorBuildNumber" type="xs:int" use="optional"/>
<xs:attribute name="MinorBuildNumber" type="xs:int" use="optional"/>
<xs:attribute name="Version" type="xs:string" use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="RequestServerVersion">
<xs:complexType>
<xs:attribute name="Version" type="t:ExchangeVersionType" fixed="Exchange2010"
use="required"/>
<xs:anyAttribute namespace="http://schemas.xmlsoap.org/soap/envelope/">
          </xs:anyAttribute>
</xs:complexType>
</xs:element>
</xs:schema>
</wsdl:types>
<wsdl:message name="GetServiceConfigurationSoapIn">
<wsdl:part name="request" element="tns:GetServiceConfiguration"/>
<wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
<wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
<wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
</wsdl:message>
<wsdl:message name="GetServiceConfigurationSoapOut">
<wsdl:part name="GetServiceConfigurationResult"
element="tns:GetServiceConfigurationResponse"/>
<wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:portType name="ExchangeServicePortType">
<wsdl:operation name="GetServiceConfiguration">
<wsdl:input message="tns:GetServiceConfigurationSoapIn"/>
<wsdl:output message="tns:GetServiceConfigurationSoapOut"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
<wsdl:documentation>
<wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0" xmlns:wsi="http://ws-
i.org/schemas/conformanceClaim/"/>
</wsdl:documentation>
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
<wsdl:operation name="GetServiceConfiguration">
<soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetServiceConfigurat
ion"/>
<wsdl:input>
<soap:body parts="request" use="literal"/>
<soap:header message="tns:GetServiceConfigurationSoapIn" part="Impersonation" use="literal"/>
```

```
<soap:header message="tns:GetServiceConfigurationSoapIn" part="RequestVersion"
use="literal"/>
<soap:header message="tns:GetServiceConfigurationSoapIn" part="MailboxCulture"
use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body parts="GetServiceConfigurationResult" use="literal"/>
<soap:header message="tns:GetServiceConfigurationSoapOut" part="ServerVersion"
use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
```

7 Appendix B: Product Behavior

The information in this specification is applicable to the following product versions. References to product versions include released service packs.

- Microsoft Exchange Server 2010

Exceptions, if any, are noted below. If a service pack number appears with the product version, behavior changed in that service pack. The new behavior also applies to subsequent service packs of the product unless otherwise specified.

Unless otherwise specified, any statement of optional behavior in this specification 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 product does not follow the prescription.

8 Change Tracking

This section identifies changes made to [MS-OXWCONFIG] protocol documentation between July 2009 and November 2009 releases. Changes are classed as major, minor, or editorial.

Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- A protocol is deprecated.
- The removal of a document from the documentation set.
- Changes made for template compliance.

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

Major and minor changes can be described further using the following revision types:

- New content added.
- Content update.
- 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.
- New content added for template compliance.
- Content updated for template compliance.

- Content removed for template compliance.
- Obsolete document removed.

Editorial changes always have the revision type "Editorially updated."

Some important terms used in revision 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.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
1.5 Prerequisites/Preconditions	49104 Added prerequisite information about the Autodiscover protocol that identifies the URL of the service.	N	New content added.
1.6 Applicability Statement	49106 Added information about when it is appropriate to use this protocol.	N	New content added.

9 Index

V

W