

[MS-OXWCONFIG]: Web Service Configuration 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 support. 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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, 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 might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications 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 this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](#).
- **Trademarks.** The names of companies and products contained in this documentation might 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, email 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 documentation 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 standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Support. For questions and support, please contact dochelp@microsoft.com.

Revision Summary

Date	Revision History	Revision Class	Comments
4/10/2009	0.1	Major	Initial Availability.
7/15/2009	1.0	Major	Revised and edited for technical content.
11/4/2009	1.1.0	Minor	Updated the technical content.
2/10/2010	2.0.0	Major	Updated and revised the technical content.
5/5/2010	3.0.0	Major	Updated and revised the technical content.
8/4/2010	3.1	Minor	Clarified the meaning of the technical content.
11/3/2010	3.2	Minor	Clarified the meaning of the technical content.
3/18/2011	3.2	None	No changes to the meaning, language, or formatting of the technical content.
8/5/2011	3.3	Minor	Clarified the meaning of the technical content.
10/7/2011	3.3	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	4.0	Major	Significantly changed the technical content.
4/27/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	4.1	Minor	Clarified the meaning of the technical content.
10/8/2012	5.0	Major	Significantly changed the technical content.
2/11/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
7/26/2013	6.0	Major	Significantly changed the technical content.
11/18/2013	6.0	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	6.0	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	6.1	Minor	Clarified the meaning of the technical content.
7/31/2014	6.1	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	6.2	Minor	Clarified the meaning of the technical content.
3/16/2015	7.0	Major	Significantly changed the technical content.
5/26/2015	7.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2015	7.0	None	No changes to the meaning, language, or formatting of the technical content.
6/13/2016	7.0	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
9/14/2016	7.0	None	No changes to the meaning, language, or formatting of the technical content.
9/19/2017	8.0	Major	Significantly changed the technical content.
7/24/2018	9.0	Major	Significantly changed the technical content.
10/1/2018	10.0	Major	Significantly changed the technical content.
4/22/2021	11.0	Major	Significantly changed the technical content.
8/17/2021	12.0	Major	Significantly changed the technical content.
2/15/2022	12.0	None	No changes to the meaning, language, or formatting of the technical content.
8/20/2024	13.0	Major	Significantly changed the technical content.

Table of Contents

1	Introduction	7
1.1	Glossary	7
1.2	References	9
1.2.1	Normative References	9
1.2.2	Informative References	9
1.3	Overview	10
1.4	Relationship to Other Protocols	10
1.5	Prerequisites/Preconditions	10
1.6	Applicability Statement	10
1.7	Versioning and Capability Negotiation	11
1.8	Vendor-Extensible Fields	11
1.9	Standards Assignments.....	11
2	Messages.....	12
2.1	Transport	12
2.2	Common Message Syntax	12
2.2.1	Namespaces	12
2.2.2	Messages.....	12
2.2.3	Elements	12
2.2.4	Complex Types.....	13
2.2.5	Simple Types	13
2.2.6	Attributes	13
2.2.7	Groups	13
2.2.8	Attribute Groups.....	13
3	Protocol Details	14
3.1	ExchangeServicePortType Server Details.....	14
3.1.1	Abstract Data Model.....	14
3.1.2	Timers	14
3.1.3	Initialization.....	14
3.1.4	Message Processing Events and Sequencing Rules	14
3.1.4.1	GetServiceConfiguration Operation.....	14
3.1.4.1.1	Messages	16
3.1.4.1.1.1	GetServiceConfigurationSoapIn Message	16
3.1.4.1.1.2	GetServiceConfigurationSoapOut Message	17
3.1.4.1.2	Elements.....	17
3.1.4.1.2.1	m:GetServiceConfiguration Element	18
3.1.4.1.2.2	m:GetServiceConfigurationResponse Element.....	18
3.1.4.1.2.3	PolicyNudgeRulesConfiguration Element.....	18
3.1.4.1.2.4	rule Element	18
3.1.4.1.2.5	mce:RulePackage Element	18
3.1.4.1.3	Complex Types	19
3.1.4.1.3.1	actionsType Complex Type.....	22
3.1.4.1.3.2	actionTypeType Complex Type	22
3.1.4.1.3.3	mce:AffinityType Complex Type.....	24
3.1.4.1.3.4	mce:AnyType Complex Type	25
3.1.4.1.3.5	ApplyType Complex Type.....	25
3.1.4.1.3.6	t:ArrayOfProtectionRulesType Complex Type	26
3.1.4.1.3.7	m:ArrayOfServiceConfigurationResponseMessageType Complex Type.....	26
3.1.4.1.3.8	m:ArrayOfServiceConfigurationType Complex Type	27
3.1.4.1.3.9	ClassificationDefinitionsType Complex Type.....	27
3.1.4.1.3.10	ClassificationDefinitionType Complex Type	28
3.1.4.1.3.11	classificationType Complex Type	29
3.1.4.1.3.12	t:ConfigurationRequestDetailsType.....	29
3.1.4.1.3.13	mce:DescriptionType Complex Type	30

3.1.4.1.3.14	mce:DetailsType Complex Type	30
3.1.4.1.3.15	mce:EncryptionType Complex Type.....	31
3.1.4.1.3.16	mce:EntityType Complex Type	31
3.1.4.1.3.17	mce:EvidenceType Complex Type	32
3.1.4.1.3.18	m:GetServiceConfigurationResponseMessageType Complex Type	33
3.1.4.1.3.19	m:GetServiceConfigurationType Complex Type.....	33
3.1.4.1.3.20	mce:GroupType Complex Type	34
3.1.4.1.3.21	mce:IdMatchType Complex Type	35
3.1.4.1.3.22	mce:KeywordType Complex Type	35
3.1.4.1.3.23	localeType Complex Type.....	36
3.1.4.1.3.24	mce:LocalizedDetailsType Complex Type	36
3.1.4.1.3.25	mce:LocalizedStringsType Complex Type	37
3.1.4.1.3.26	mce:MatchType Complex Type	38
3.1.4.1.3.27	t:MailTipsServiceConfiguration Complex Type.....	38
3.1.4.1.3.28	orType Complex Type.....	40
3.1.4.1.3.29	mce:PatternType Complex Type	40
3.1.4.1.3.30	t:PolicyNudgeRulesServiceConfiguration Complex Type.....	41
3.1.4.1.3.31	PolicyNudgeRulesConfigurationType Complex Type	41
3.1.4.1.3.32	mce:PolicyNudgeRulesType Complex Type	42
3.1.4.1.3.33	PolicyNudgeRuleType Complex Type.....	42
3.1.4.1.3.34	predicateElementConstantType Complex Type	43
3.1.4.1.3.35	predicateElementMultipleChildType Complex Type.....	43
3.1.4.1.3.36	predicateElementSingleChildType Complex Type.....	44
3.1.4.1.3.37	t:ProtectionRuleActionType Complex Type	45
3.1.4.1.3.38	t:ProtectionRuleAndType Complex Type.....	46
3.1.4.1.3.39	t:ProtectionRuleArgumentType Complex Type	47
3.1.4.1.3.40	t:ProtectionRuleConditionType Complex Type	47
3.1.4.1.3.41	t:ProtectionRuleRecipientIsType Complex Type.....	48
3.1.4.1.3.42	t:ProtectionRuleSenderDepartmentsType Complex Type	49
3.1.4.1.3.43	t:ProtectionRulesServiceConfiguration Complex Type	49
3.1.4.1.3.44	t:ProtectionRuleType Complex Type	50
3.1.4.1.3.45	mce:PublisherType Complex Type	51
3.1.4.1.3.46	recipientType Complex Type.....	51
3.1.4.1.3.47	mce:RegexType Complex Type	52
3.1.4.1.3.48	mce:ResourceNameType Complex Type.....	52
3.1.4.1.3.49	mce:ResourceType Complex Type	53
3.1.4.1.3.50	mce:RulePackageContainerType Complex Type	54
3.1.4.1.3.51	mce:RulePackageType Complex Type	54
3.1.4.1.3.52	mce:RulePackType Complex Type	55
3.1.4.1.3.53	ruleType Complex Type	56
3.1.4.1.3.54	mce:RulesType Complex Type	56
3.1.4.1.3.55	senderType Complex Type	57
3.1.4.1.3.56	t:ServiceConfiguration Complex Type	57
3.1.4.1.3.57	m:ServiceConfigurationResponseMessageType Complex Type	58
3.1.4.1.3.58	t:SmtpDomain Complex Type	59
3.1.4.1.3.59	t:SmtpDomainList Complex Type	59
3.1.4.1.3.60	mce:TermType Complex Type	59
3.1.4.1.3.61	t:UnifiedMessageServiceConfiguration Complex Type	60
3.1.4.1.3.62	VersionType Complex Type	61
3.1.4.1.3.63	versionType Complex Type	61
3.1.4.1.3.64	mce:VersionedRuleType Complex Type.....	62
3.1.4.1.3.65	mce:VersionedPatternType Complex Type.....	62
3.1.4.1.3.66	mce:VersionedEvidenceType Type	63
3.1.4.1.3.67	mce:FingerprintType Type	63
3.1.4.1.4	Simple Types	64
3.1.4.1.4.1	mce:GuidType Simple Type.....	65
3.1.4.1.4.2	mce:LangType Simple Type	65
3.1.4.1.4.3	minRequiredVersionType Simple Type	66

3.1.4.1.4.4	mce:NameType Simple Type	66
3.1.4.1.4.5	mce:OptionalNameType Simple Type	66
3.1.4.1.4.6	t:ProtectionRuleActionKindType Simple Type	66
3.1.4.1.4.7	t:ProtectionRuleAllInternalType Simple Type	67
3.1.4.1.4.8	t:ProtectionRuleTrueType Simple Type	67
3.1.4.1.4.9	t:ProtectionRuleValueType Simple Type	67
3.1.4.1.4.10	mce:ProbabilityType Simple Type	67
3.1.4.1.4.11	mce:ProximityType Simple Type	68
3.1.4.1.4.12	mce:RestrictedTermType Simple Type	68
3.1.4.1.4.13	mce:RulePackNameType Simple Type.....	68
3.1.4.1.4.14	t:ServiceConfigurationType Simple Type.....	68
3.1.4.1.4.15	mce:WorkloadType Simple Type	69
3.1.4.1.4.16	yesnoType Simple Type	69
3.1.4.1.4.17	mce:EngineVersionType Simple Type	69
3.1.4.1.4.18	mce:FingerprintValueType Simple Type	70
3.1.4.1.5	Attributes	70
3.1.4.1.6	Groups.....	70
3.1.4.1.7	Attribute Groups.....	70
3.1.5	Timer Events.....	70
3.1.6	Other Local Events.....	70
4	Protocol Examples	71
4.1	GetServiceConfiguration Operation Request.....	71
4.2	GetServiceConfiguration Operation Response.....	71
4.3	Unsuccessful GetServiceConfiguration Operation Response.....	72
4.3.1	SOAP Exception.....	72
4.3.2	GetServiceConfiguration Operation Error Response	73
5	Security	74
5.1	Security Considerations for Implementers	74
5.2	Index of Security Parameters	74
6	Appendix A: Full WSDL	75
7	Appendix B: Full XML Schema.....	77
7.1	Classification Rule Package Container Type Schema	77
7.2	DLP Rule Schema	77
7.3	DLP Rule Types Schema.....	78
7.4	DLP Rules Configuration Schema.....	80
7.5	DLP Rules Configuration Types Schema	81
7.6	Messages Schema	82
7.7	Rule Package Schema.....	84
7.8	Rule Package Types Schema.....	84
7.9	Types Schema	89
8	Appendix C: Product Behavior	93
9	Change Tracking.....	98
10	Index.....	99

1 Introduction

The Web Service Configuration Protocol sends the request-response messages that retrieve configuration and policy information for a **mailbox**.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

domain: A set of users and computers sharing a common namespace and management infrastructure. At least one computer member of the set has to act as a domain controller (DC) and host a member list that identifies all members of the domain, as well as optionally hosting the Active Directory service. The domain controller provides authentication of members, creating a unit of trust for its members. Each domain has an identifier that is shared among its members. For more information, see [\[MS-AUTHSOD\]](#) section 1.1.1.5 and [\[MS-ADTS\]](#).

email address: A string that identifies a user and enables the user to receive Internet messages.

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 [\[C706\]](#) have to be used for generating the GUID. See also universally unique identifier (UUID).

Hypertext Transfer Protocol (HTTP): An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

Hypertext Transfer Protocol Secure (HTTPS): An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [\[SSL3\]](#) and [\[RFC5246\]](#).

mail tip: A note that is presented to the author of a message when the author is composing the message. A mail tip provides information about the recipients of a message and issues that might impact delivery of the message, such as moderation or delivery restrictions.

mailbox: A message store that contains email, calendar items, and other **Message objects** for a single recipient.

Message object: A set of properties that represents an email message, appointment, contact, or other type of personal-information-management object. In addition to its own properties, a Message object contains recipient properties that represent the addressees to which it is addressed, and an attachments table that represents any files and other Message objects that are attached to it.

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 email client. The action part specifies what action the email client takes if the condition is met. Organization policies are sets of conditions and associated actions that apply within an organization.

policy tip: A message that is displayed in the client user interface to inform the user that an email does not comply with an email policy configured on the server.

recipient: An entity that can receive email messages.

rule: An item that defines a condition and an action. The condition is evaluated for each **Message object** as it is delivered, and the action is executed if the new Message object matches the condition.

Simple Mail Transfer Protocol (SMTP): A member of the TCP/IP suite of protocols that is used to transport Internet messages, as described in [\[RFC5321\]](#).

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 action: The HTTP request header field used to indicate the intent of the **SOAP** request, using a URI value. See [\[SOAP1.1\]](#) section 6.1.1 for more information.

SOAP fault: A container for error and status information within a SOAP message. See [\[SOAP1.2-1/2007\]](#) section 5.4 for more information.

Unified Messaging: A set of components and services that enable voice, fax, and email messages to be stored in a user's **mailbox** and accessed from a variety of devices.

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [\[RFC1738\]](#).

web server: A server computer that hosts websites and responds to requests from applications.

Web Services Description Language (WSDL): An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

WSDL message: An abstract, typed definition of the data that is communicated during a **WSDL operation** [\[WSDL\]](#). Also, an element that describes the data being exchanged between web service providers and clients.

WSDL operation: A single action or function of a web service. The execution of a WSDL operation typically requires the exchange of messages between the service requestor and the service provider.

WSDL port type: A named set of logically-related, abstract **Web Services Description Language (WSDL)** operations and messages.

XML: The Extensible Markup Language, as described in [\[XML1.0\]](#).

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 [\[XMLNS-2ED\]](#).

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

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.

[MS-OXWMT] Microsoft Corporation, "[Mail Tips Web Service Extensions](#)".

[MS-OXWSCDATA] Microsoft Corporation, "[Common Web Service Data Types](#)".

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

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <https://www.rfc-editor.org/info/rfc2616>

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, <https://www.rfc-editor.org/info/rfc2818>

[RFC3066] Alvestrand, H., "Tags for the Identification of Languages", BCP 47, RFC 3066, January 2001, <https://www.rfc-editor.org/info/rfc3066>

[RFC4646] Phillips, A., and Davis, M., Eds., "Tags for Identifying Languages", BCP 47, RFC 4646, September 2006, <https://www.rfc-editor.org/info/rfc4646>

[SOAP1.1] Box, D., Ehnebuske, D., Kaktivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <https://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, <https://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <https://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <https://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, <https://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

1.2.2 Informative References

[MS-OXPROTO] Microsoft Corporation, "[Exchange Server Protocols System Overview](#)".

[MS-OXWSADISC] Microsoft Corporation, "[Autodiscover Publishing and Lookup SOAP-Based Web Service Protocol](#)".

1.3 Overview

The Web Service Configuration Protocol sends the request-response messages that retrieve **organization policy** configuration information for a **mailbox**. This includes configuration information for the following:

- **Mail tips**
- Protection rules – Protection rules are a collection of **rules** that protect email messages by applying a rights protection template to the message.
- Data loss prevention (DLP) **policy tips** – DLP is a collection of features that provides content-aware classification and applies an organization policy to sensitive data within an email message.
- **Unified Messaging**

A client uses the **GetServiceConfiguration WSDL operation**, as described in section [3.1.4.1](#), to view the organization policy information for a user. An example of how to retrieve the information and examples of successful and unsuccessful responses are included in section [4](#).

Clients use **SOAP**, as described in [\[SOAP1.1\]](#), to contact the Web Service Configuration Protocol.

1.4 Relationship to Other Protocols

The Web Service Configuration Protocol uses **SOAP** over **HTTP** and SOAP over **HTTPS**, as shown in the following layering diagram.

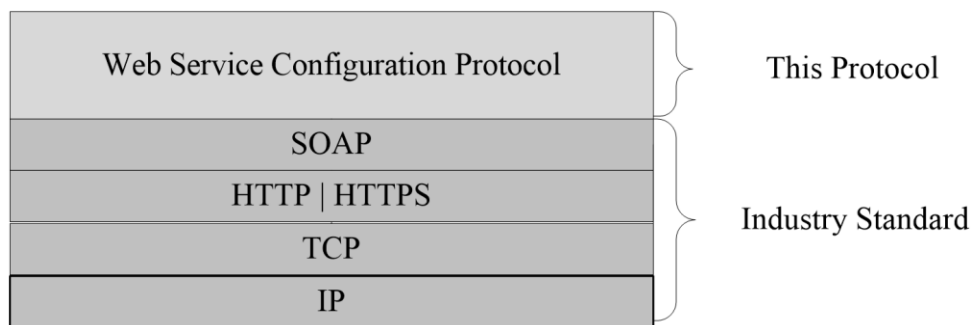


Figure 1: This protocol in relation to other protocols

For conceptual background information and overviews of the relationships and interactions between this and other protocols, see [\[MS-OXPROTO\]](#).

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, as described in [\[MS-OXWSADISC\]](#).

This protocol relies on the **web server** that hosts the application to perform authentication.

1.6 Applicability Statement

This protocol accesses configuration information for an **organization policy** when that 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 described in section [2.1](#).
- **Protocol Versions:** This protocol specifies only one **WSDL port type** version. The **RequestVersion** element of the **GetServiceConfigurationSoapIn WSDL message**, as described in section [3.1.4.1.1.1](#), identifies the WSDL port type version of the request. The **ServerVersion** element of the **GetServiceConfigurationSoapOut** WSDL message, as described in section [3.1.4.1.1.2](#), identifies the version of the server responding to the request.
- **Security and Authentication Methods:** This protocol relies on the **web server** that is hosting it to perform authentication.
- **Localization:** This protocol uses the **MailboxCulture** element of the **GetServiceConfigurationSoapIn** WSDL message to specify the culture of a **mailbox**.
- **Capability Negotiation:** This protocol does not support version negotiation.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The **WSDL** in this specification provides a base description of the protocol. The schema in this specification provides a base description of the message syntax. The text that specifies the WSDL and schema might specify restrictions that reflect actual 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**, **not null**, or **present**.

2.1 Transport

This protocol uses **SOAP** version 1.1, as specified in [\[SOAP1.1\]](#).

This protocol MUST support SOAP over **HTTP**, as specified in [\[RFC2616\]](#). This protocol SHOULD use secure communications via **HTTPS**, as specified in [\[RFC2818\]](#).

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	
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
wsdl	http://schemas.xmlsoap.org/WSDL/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	
m	http://schemas.microsoft.com/exchange/services/2006/messages	
mce	http://schemas.microsoft.com/office/2011/mce	

2.2.2 Messages

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

2.2.3 Elements

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

2.2.4 Complex Types

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

2.2.5 Simple Types

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

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

3 Protocol Details

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.

In the following sections, the schema definition might be less restrictive than the processing rules imposed by the protocol. The **WSDL** in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL specifies additional restrictions that reflect actual Microsoft product 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, not null and present.

3.1 ExchangeServicePortType Server Details

The Web Service Configuration Protocol defines a single WSDL port type with one **WSDL operation**, which gets the service configuration for a **mailbox**.

3.1.1 Abstract Data Model

None.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

The following table summarizes the list of operations as defined by this specification.

Operation	Description
GetServiceConfiguration (section 3.1.4.1)	Gets information about the organization policy configuration for a mailbox .

3.1.4.1 GetServiceConfiguration Operation

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

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

The following is the **WSDL** binding specification for the **GetServiceConfiguration WSDL operation**.

```
<wsdl:operation name="GetServiceConfiguration">
```

```

<soap:operation
  soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/
  GetServiceConfiguration"/>
<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>

```

A successful **GetServiceConfiguration** WSDL operation request returns a **GetServiceConfigurationResponse** element, as specified in section [3.1.4.1.2.2](#), with the **ResponseClass** attribute, as defined in [\[MS-OXWSCDATA\]](#) section 2.2.4.65, set to "Success". The **ResponseCode** element of the **GetServiceConfigurationResponse** element, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.65, is set to "NoError".

If the **GetServiceConfiguration** WSDL operation is not successful, the operation returns a **GetServiceConfigurationResponse** element with the **ResponseClass** attribute set to "Error". The **ResponseCode** element of the **GetServiceConfigurationResponse** element is set to one of the common errors defined in [\[MS-OXWSCDATA\]](#) section 2.2.5.24.

The **GetServiceConfiguration** WSDL operation MUST use the **SOAP fault** specified in this section if the request **XML** is malformed according the schema specified in sections [6](#) and [7](#). The following XML specifies the structure of the SOAP fault that is returned when a request contains malformed XML. This message is not specified in the schema.

```

<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>

```

The **faultcode** element MUST have a text value of "a:ErrorSchemaValidation".

The **faultstring** element contains a human-readable explanation of the fault.

The **detail** element contains application-specific error information. This element is specified by the following child elements:

- **ResponseCode:** This element MUST have a text value of "ErrorSchemaValidation".
- **Message:** This element MUST have a text value of "The request failed schema validation."
- **MessageXML:** This element contains additional error response information. This information includes the line number of the error, specified by the **LineNumber** element; the line position of the error, specified by the **LinePosition** element; and a human-readable explanation of the fault, specified by the **Violation** element.

3.1.4.1.1 Messages

The following **WSDL message** definitions are specific to the **GetServiceConfiguration WSDL operation**, as specified in section [3.1.4.1](#).

Message	Description
GetServiceConfigurationSoapIn (section 3.1.4.1.1.1)	Specifies the request to get the service configuration.
GetServiceConfigurationSoapOut (section 3.1.4.1.1.2)	Specifies the response to the GetServiceConfigurationSoapIn request WSDL message.

3.1.4.1.1.1 GetServiceConfigurationSoapIn Message

The **GetServiceConfigurationSoapIn WSDL message** specifies the request to get the service configuration.

The following is the **GetServiceConfigurationSoapIn** WSDL message specification.

```

<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>

```


The **GetServiceConfigurationSoapIn** WSDL message is the input message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/GetServiceConfiguration>.

The parts of the **GetServiceConfigurationSoapIn** WSDL message are described in the following table.

Part name	Element/type	Description
request	m:GetServiceConfiguration (section 3.1.4.1.2.1)	Specifies the request.
Impersonation	t:ExchangeImpersonation ([MS-OXWSCDATA] section 2.2.3.3)	Specifies the account to impersonate.
RequestVersion	t:RequestServerVersion ([MS-OXWSCDATA] section 2.2.3.9)	Specifies the schema version for the GetServiceConfiguration WSDL operation request (section 3.1.4.1).
MailboxCulture	t:MailboxCulture ([MS-OXWSCDATA] section 2.2.3.6)	Specifies the culture to use for accessing the mailbox . The cultures are defined in [RFC3066] .

3.1.4.1.1.2 GetServiceConfigurationSoapOut Message

The **GetServiceConfigurationSoapOut WSDL message** specifies the response to the **GetServiceConfigurationSoapIn** request WSDL message, as specified in section [3.1.4.1.1.1](#).

The following is the **GetServiceConfigurationSoapOut** WSDL message specification.

```
<wsdl:message name="GetServiceConfigurationSoapOut">
  <wsdl:part name="GetServiceConfigurationResult"
    element="tns:GetServiceConfigurationResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
```

The **GetServiceConfigurationSoapOut** WSDL message is the output message for the **SOAP action** <http://schemas.microsoft.com/exchange/services/2006/messages/GetServiceConfiguration>.

The parts of the **GetServiceConfigurationSoapOut** WSDL message are described in the following table.

Part name	Element/type	Description
GetServiceConfigurationResult	m:GetServiceConfigurationResponse (section 3.1.4.1.2.2)	Specifies the response.
ServerVersion	t:ServerVersionInfo ([MS-OXWSCDATA] section 2.2.3.10)	Specifies the server version for the response.

3.1.4.1.2 Elements

The following table lists the **XML schema** elements that are specific to the **GetServiceConfiguration WSDL operation**, as specified in section [3.1.4.1](#).

Element	Description
GetServiceConfiguration (section 3.1.4.1.2.1)	Specifies the base element for a GetServiceConfiguration WSDL operation request.
GetServiceConfigurationResponse (section 3.1.4.1.2.2)	Specifies the response message for a GetServiceConfiguration WSDL operation.
PolicyNudgeRulesConfiguration (section 3.1.4.1.2.3)	Specifies a set of DLP rules and rule classification definitions.
rule (section 3.1.4.1.2.4)	Specifies a DLP rule.
RulePackage (section 3.1.4.1.2.5)	Specifies the root of the rule pack document (section 3.1.4.1.3.52).

3.1.4.1.2.1 m:GetServiceConfiguration Element

The **GetServiceConfiguration** element specifies the base element for a **GetServiceConfiguration WSDL operation** request, as specified in section [3.1.4.1](#).

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

3.1.4.1.2.2 m:GetServiceConfigurationResponse Element

The **GetServiceConfigurationResponse** element specifies the response message for a **GetServiceConfiguration WSDL operation**, as specified in section [3.1.4.1](#).

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

3.1.4.1.2.3 PolicyNudgeRulesConfiguration Element

The **PolicyNudgeRulesConfiguration** element specifies a set of DLP rules and rule classification definitions. [<1>](#)

```
<xs:element name="PolicyNudgeRulesConfiguration"
type="PolicyNudgeRulesConfigurationType" />
```

3.1.4.1.2.4 rule Element

The **rule** element specifies a DLP rule. [<2>](#)

```
<xs:element name="rule" type="ruleType"/>
```

3.1.4.1.2.5 mce:RulePackage Element

The **RulePackage** element specifies the root of the rule pack document. [<3>](#) A rule pack is a collection of **rules** and associated resources, as specified in section [3.1.4.1.3.52](#).

```
<xs:element name="RulePackage" type="mce:RulePackageType"/>
```

3.1.4.1.3 Complex Types

The following table summarizes the **XML schema** complex type definitions that are specific to the **GetServiceConfiguration WSDL operation**, as specified in section [3.1.4.1](#).

Complex type	Description
actionsType (section 3.1.4.1.3.1)	Specifies the actions that a DLP rule can have.
actionTypeType (section 3.1.4.1.3.2)	Specifies the following items: <ul style="list-style-type: none"> An optional URL to a compliance note A policy tip message Whether the classification rule can be overridden
AffinityType (section 3.1.4.1.3.3)	Specifies an affinity-based classification type.
AnyType (section 3.1.4.1.3.4)	Specifies a classification rule where, if any of the rule conditions are met, the rule condition is processed as a match.
ApplyType (section 3.1.4.1.3.5)	Specifies whether the client has to update its email policy rule set.
ArrayOfProtectionRulesType (section 3.1.4.1.3.6)	Specifies an array of protection rules.
ArrayOfServiceConfigurationResponseMessageType (section 3.1.4.1.3.7)	Specifies an array of service configuration response messages.
ArrayOfServiceConfigurationType (section 3.1.4.1.3.8)	Specifies the requested service configurations for a GetServiceConfigurationSoapIn WSDL message (section 3.1.4.1.1.1).
ClassificationDefinitionsType (section 3.1.4.1.3.9)	Specifies the definitions used to classify messages.
ClassificationDefinitionType (section 3.1.4.1.3.10)	Specifies a single message classification definition.
classificationType (section 3.1.4.1.3.11)	Specifies a classification identifier.
ConfigurationRequestDetailsType (section 3.1.4.1.3.12)	Contains a request for details of the policy tips .
DescriptionType (section 3.1.4.1.3.13)	Specifies the description of a localized resource string.
DetailsType (section 3.1.4.1.3.14)	Specifies the localized description of a rule pack (section 3.1.4.1.3.52).
EncryptionType (section 3.1.4.1.3.15)	Specifies the encryption key and initialization vector.
EntityType (section 3.1.4.1.3.16)	Specifies an entity classification type.
EvidenceType (section 3.1.4.1.3.17)	Specifies the classification rules for an affinity-based classification.
GetServiceConfigurationResponseMessageType (section 3.1.4.1.3.18)	Contains the response message for a GetServiceConfiguration WSDL operation.

Complex type	Description
GetServiceConfigurationType (section 3.1.4.1.3.19)	Specifies the requested service configurations and identifies the sender or impersonated user who is making the request.
GroupType (section 3.1.4.1.3.20)	Specifies the terms in a keyword-based rule.
IdMatchType (section 3.1.4.1.3.21)	Specifies an entity-based primary match rule.
KeywordType (section 3.1.4.1.3.22)	Specifies the terms and identifier of a keyword-based rule.
localeType (section 3.1.4.1.3.23)	Specifies the unique identifier of a keyword-based text processor.
LocalizedDetailsType (section 3.1.4.1.3.24)	Specifies the localized information about a rule pack.
LocalizedStringsType (section 3.1.4.1.3.25)	Specifies localized information about classification rules for both entity-based and affinity-based rules.
MatchType (section 3.1.4.1.3.26)	Specifies a classification rule match.
MailTipsServiceConfiguration (section 3.1.4.1.3.27)	Contains service configuration information for the mail tips service.
orType (section 3.1.4.1.3.28)	Specifies a logical OR for rule predicates.
PatternType (section 3.1.4.1.3.29)	Specifies the pattern for an entity classification.
PolicyNudgeRulesServiceConfiguration (section 3.1.4.1.3.30)	Contains the policy tip configuration data. <4>
PolicyNudgeRulesConfigurationType (section 3.1.4.1.3.31)	Specifies the set of DLP rules and classification definitions that are sent to a client.
PolicyNudgeRulesType (section 3.1.4.1.3.32)	Specifies a collection of DLP rules.
PolicyNudgeRuleType (section 3.1.4.1.3.33)	Specifies a single DLP rule.
predicateElementConstantType (section 3.1.4.1.3.34)	Specifies the type of all constant rule predicates. All elements of this type MUST be empty.
predicateElementMultipleChildType (section 3.1.4.1.3.35)	Specifies multiple predicates and logical operators for a classification definition.
predicateElementSingleChildType (section 3.1.4.1.3.36)	Specifies a single predicate or logical operator for a classification definition.
ProtectionRuleActionType (section 3.1.4.1.3.37)	Specifies the action that the client can take if the condition part of the associated rule matches.
ProtectionRuleAndType (section 3.1.4.1.3.38)	Specifies that there MUST be more than one protection rule condition.
ProtectionRuleArgumentType (section 3.1.4.1.3.39)	Specifies an attribute that is used to specify an argument to an action.
ProtectionRuleConditionType (section 3.1.4.1.3.40)	Specifies the condition part of a protection rule.

Complex type	Description
ProtectionRuleRecipientIsType (section 3.1.4.1.3.41)	Specifies a condition that matches if any recipients of the email message match any specified recipients in the child Value elements.
ProtectionRuleSenderDepartmentsType (section 3.1.4.1.3.42)	Specifies a condition that matches if the department of the sender of the email message matches any of the specified departments in the child Value elements.
ProtectionRulesServiceConfiguration (section 3.1.4.1.3.43)	Specifies the configuration of the protection rules service.
ProtectionRuleType (section 3.1.4.1.3.44)	Specifies a single protection rule.
PublisherType (section 3.1.4.1.3.45)	Specifies the unique identifier of a rule pack publisher.
recipientType (section 3.1.4.1.3.46)	Specifies a recipient in a classification definition. No more than one of the attributes in this type can be present.
RegexType (section 3.1.4.1.3.47)	Specifies a regular expression text processor.
ResourceNameType (section 3.1.4.1.3.48)	Specifies a localized name for a classification rule.
ResourceType (section 3.1.4.1.3.49)	Specifies the localized names and descriptions of a classification rule.
RulePackageContainerType (section 3.1.4.1.3.50)	Contains a single rule package.
RulePackageType (section 3.1.4.1.3.51)	Specifies the rule pack identifier and the set of classification definitions rules.
RulePackType (section 3.1.4.1.3.52)	Specifies the version, identifier, and publisher of a rule pack.
ruleType (section 3.1.4.1.3.53)	Specifies a DLP rule.
RulesType (section 3.1.4.1.3.54)	Specifies the rules collection in a rules package.
senderType (section 3.1.4.1.3.55)	Specifies a sender in a classification definition.
ServiceConfiguration (section 3.1.4.1.3.56)	Specifies the base type for the service configuration types.
ServiceConfigurationResponseMessageType (section 3.1.4.1.3.57)	Specifies service configuration settings.
SmtpDomain (section 3.1.4.1.3.58)	Specifies a single domain .
SmtpDomainList (section 3.1.4.1.3.59)	Specifies a list of internal domains in a user's organization.
TermType (section 3.1.4.1.3.60)	Specifies a term and whether the term is case-sensitive.
UnifiedMessageServiceConfiguration (section 3.1.4.1.3.61)	Specifies the configuration for the Unified Messaging service.
VersionType (section 3.1.4.1.3.62)	Specifies the rule pack version.

Complex type	Description
versionType (section 3.1.4.1.3.63)	Specifies the minimum client version that can use this rule and the rule conditions and actions.
VersionedRuleType (section 3.1.4.1.3.64)	Specifies the required classification engine version for a list of data classification rules.
VersionedPatternType (section 3.1.4.1.3.65)	Specifies the required classification engine version for a list of patterns.
VersionedEvidenceType (section 3.1.4.1.3.66)	Specifies the required classification engine version for a list of evidences.
FingerprintType (section 3.1.4.1.3.67)	Specifies the fingerprint definition for a classification rule.

3.1.4.1.3.1 actionsType Complex Type

The **actionsType** complex type specifies the actions that a DLP rule can have. [<5>](#)

```
<xs:complexType name="actionsType">
  <xs:choice maxOccurs="unbounded">
    <xs:element name="block" type="actionTypeType" />
    <xs:element name="notify" type="actionTypeType" />
  </xs:choice>
</xs:complexType>
```

The following table lists the child elements of the **actionsType** complex type.

Element	Type	Description
block	actionTypeType (section 3.1.4.1.3.2)	Specifies that if a rule condition is met, the server MUST block sending the message unless it is overridden by user input.
notify	actionTypeType	Specifies that if a rule condition is met, the client MUST notify the user about the policy violation in the form of a policy tip .

3.1.4.1.3.2 actionTypeType Complex Type

The **actionTypeType** complex type specifies the following items:

- An optional URL to a compliance note.
- A policy tip message.
- Whether the classification rule can be overridden. [<6>](#)

```
<xs:complexType name="actionTypeType">
  <xs:sequence>
    <xs:element name="message">
```

```

<xs:complexType>
  <xs:sequence minOccurs="1" maxOccurs="unbounded">
    <xs:element name="locale" type="localeType" />
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="override">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="justification">
        <xs:complexType>
          <xs:attribute name="type" use="required">
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:enumeration value="none" />
                <xs:enumeration value="optional" />
                <xs:enumeration value="required" />
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="allow" type="yesnoType" use="required" />
  </xs:complexType>
</xs:element>
<xs:element name="falsePositive">
  <xs:complexType>
    <xs:attribute name="allow" type="yesnoType" use="required" />
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **actionTypeType** complex type.

Element	Type	Description
message		Contains the locale child element.
locale	localeType (section 3.1.4.1.3.23)	Specifies the local natural language identifier, an optional URL to a compliance note, policy tip messages, and an override text message.
override		Contains the justification child element and the allow attribute.
justification		Contains the type attribute.
falsePositive		Contains the allow attribute.

The following table lists the attributes of the **actionTypeType** complex type.

Attribute	Type	Description
type		Specifies whether an override justification is required. This attribute MUST be present.
allow	yesnoType (section 3.1.4.1.4.16)	In the override element, this attribute specifies whether a rule can be overridden. This attribute

Attribute	Type	Description
		MUST be present. The client MUST ignore this value if the notify element (section 3.1.4.1.3.1) is present. In the falsePositive element, this attribute specifies whether a false positive reporting is allowed. This attribute MUST be present.

3.1.4.1.3.3 mce:AffinityType Complex Type

The **AffinityType** complex type specifies an affinity-based classification type. [<7>](#)

```
<xs:complexType name="AffinityType">
  <xs:sequence>
    <xs:element name="Evidence" type="mce:EvidenceType" maxOccurs="unbounded"/>
    <xs:element name="Version" type="mce:VersionedEvidenceType" minOccurs="0"
maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
  <xs:attribute name="evidencesProximity" type="mce:ProximityType" use="required"/>
  <xs:attribute name="thresholdConfidenceLevel" type="mce:ProbabilityType" use="required"/>
  <xs:attribute name="workload" type="mce:WorkloadType"/>
</xs:complexType>
```

The following table lists the child elements of the **AffinityType** complex type.

Element	Type	Description
Evidence	mce:EvidenceType (section 3.1.4.1.3.17)	Specifies the rules that are matched for an affinity-based classification.
Version	mce:VersionedEvidenceType (section 3.1.4.1.3.66)	Specified the version of a list of evidences.

The following table lists the attributes of the **AffinityType** complex type.

Attribute	Type	Description
id	mce:GuidType (section 3.1.4.1.4.1)	Specifies the rule identifier. This attribute MUST be present.
evidencesProximity	mce:ProximityType (section 3.1.4.1.4.11)	Specifies the proximity value used by the classification engine. This attribute MUST be present.
thresholdConfidenceLevel	mce:ProbabilityType (section 3.1.4.1.4.10)	Specifies the confidence level threshold used by the classification engine to trigger an action. This attribute MUST be present.
workload	mce:WorkloadType (section 3.1.4.1.4.15)	This attribute is not used.

3.1.4.1.3.4 mce:AnyType Complex Type

The **AnyType** complex type specifies a classification rule where, if any of the rule conditions are met, then the rule condition is processed as a match. [<8>](#)

```
<xs:complexType name="AnyType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="Match" type="mce:MatchType"/>
      <xs:element name="Any" type="mce:AnyType"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="minMatches" type="xs:nonNegativeInteger" default="1"/>
  <xs:attribute name="maxMatches" type="xs:nonNegativeInteger" use="optional"/>
</xs:complexType>
```

The following table lists the child elements of the **AnyType** complex type.

Element	Type	Description
Match	mce:MatchType (section 3.1.4.1.3.26)	Specifies a single classification rule to match. This element can occur zero or more times.
Any	mce:AnyType (section 3.1.4.1.3.4)	Specifies two or more classification rules to match. This element can occur zero or more times.

The following table lists the attributes of the **AnyType** complex type.

Attribute	Type	Description
minMatches	xs:nonNegativeInteger ([XMLSCHEMA2] section 3.3.20)	Specifies the minimum number of rule matches required to indicate that a rule condition has been met. The default value is 1.
maxMatches	xs:nonNegativeInteger	Specifies the maximum number of rule matches required to indicate that a rule condition has been met.

3.1.4.1.3.5 ApplyType Complex Type

The **ApplyType** complex type specifies whether the client has to update its email policy rule set. [<9>](#)

```
<xs:complexType name="ApplyType">
  <xs:attribute name="apply" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="((([Tt][Rr][Uu][Ee])|([Ff][Aa][Ll][Ss][Ee])))" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
```

The following table lists the attribute of the **ApplyType** complex type.

Attribute	Type	Description
apply	xs:simpleType ([XMLSCHEMA2] section 3.2.2)	Specifies whether the client can update its rule set. If the value is "false", the client MUST NOT update its rule set. If the value is "true", the client MUST update rule set. If the value is "true" and the PolicyNudgeRules element (section 3.1.4.1.3.31) or the ClassificationDefinitions element (section 3.1.4.1.3.31) is empty, the client MUST remove all rules. The possible values for the apply attribute are "true" and "false." The values are not case sensitive.

3.1.4.1.3.6 t:ArrayOfProtectionRulesType Complex Type

The **ArrayOfProtectionRulesType** complex 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:sequence>
</xs:complexType>
```

The following table lists the child element of the **ArrayOfProtectionRulesType** complex type.

Element	Type	Description
Rule	t:ProtectionRuleType (section 3.1.4.1.3.44)	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.

3.1.4.1.3.7 m:ArrayOfServiceConfigurationResponseMessageType Complex Type

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

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

The following table lists the child element of the **ArrayOfServiceConfigurationResponseMessageType** complex type.

Element	Type	Description
ServiceConfigurationResponseMessageType	m:ServiceConfigurationResponseMessageType (section 3.1.4.1.3.57)	Contains a service configuration response message. This element MUST occur at least once and can occur two or more times.

3.1.4.1.3.8 m:ArrayOfServiceConfigurationType Complex Type

The **ArrayOfServiceConfigurationType** complex type specifies the requested service configurations for a **GetServiceConfigurationSoapIn WSDL message**, as specified in section [3.1.4.1.1.1](#).

```
<xs:complexType name="ArrayOfServiceConfigurationType">
  <xs:choice
    maxOccurs="unbounded"
    minOccurs="1"
  >
    <xs:element name="ConfigurationName"
      type="t:ServiceConfigurationType"
    />
  </xs:choice>
</xs:complexType>
```

The following table lists the child element of the **ArrayOfServiceConfigurationType** complex type.

Element	Type	Description
ConfigurationName	t:ServiceConfigurationType (section 3.1.4.1.4.14)	Specifies the service configuration that is returned in the response. This simple type MUST occur at least once.

3.1.4.1.3.9 ClassificationDefinitionsType Complex Type

The **ClassificationDefinitionsType** complex type specifies the definitions used to classify messages. This type extends the **ApplyType** complex type, as specified in section [3.1.4.1.3.5](#). [<10>](#)

```
<xs:complexType name="ClassificationDefinitionsType">
  <xs:complexContent>
    <xs:extension base="ApplyType">
      <xs:sequence>
        <xs:element name="ClassificationDefinition"
          type="ClassificationDefinitionType"
        />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

        minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

The following table describes the child element of the **ClassificationDefinitionsType** complex type.

Element	Type	Description
ClassificationDefinition	ClassificationDefinitionType (section 3.1.4.1.3.10)	Specifies a single rule package. This element can occur zero or more times. If this element is not present and the apply attribute (section 3.1.4.1.3.5) is set to "true", the client MUST delete all classification definitions. If this element is present, the client MUST apply all the classification definitions.

3.1.4.1.3.10 ClassificationDefinitionType Complex Type

The **ClassificationDefinitionType** complex type specifies a single message classification definition. This type extends the **RulePackageContainerType** complex type, as specified in section [3.1.4.1.3.50.<11>](#)

```

<xs:complexType name="ClassificationDefinitionType">
  <xs:complexContent>
    <xs:extension base="mce:RulePackageContainerType">
      <xs:attribute name="id" type="xs:string" use="required" />
      <xs:attribute name="version" type="xs:long" use="required" />
      <xs:anyAttribute processContents="skip" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The following table lists the attributes of the **ClassificationDefinitionType** complex type.

Attribute	Type	Description
id	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies the classification definition identifier. This attribute MUST be present.
version	xs:long ([XMLSCHEMA2] section 3.3.16)	Specifies the classification definition version. This attribute MUST be present.
	xs:anyAttribute ([XMLSCHEMA1] section 3.10)	Indicates that the server can send non-predefined attributes in a response to the client. The client MUST send back the same attributes in subsequent requests.

3.1.4.1.3.11 classificationType Complex Type

The **classificationType** complex type specifies a classification identifier. [<12>](#)

```
<xs:complexType name="classificationType">
  <xs:attribute name="rulePackId" type="xs:string" use="required" />
  <xs:attribute name="id" type="xs:string" use="required" />
  <xs:attribute name="minCount" type="xs:integer" use="optional" />
  <xs:attribute name="maxCount" type="xs:integer" use="optional" />
  <xs:attribute name="minConfidence" type="xs:integer" use="optional" />
  <xs:attribute name="maxConfidence" type="xs:integer" use="optional" />
</xs:complexType>
```

The following table lists the attributes of the **classificationType** complex type.

Attribute	Type	Description
rulePackId	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies the rule pack identifier. This attribute MUST be present.
id	xs:string	Specifies the classification identifier. This attribute MUST be present.
minCount	xs:integer ([XMLSCHEMA2] section 3.3.13)	Specifies the minimum count. This attribute is optional.
maxCount	xs:integer	Specifies the maximum count. This attribute is optional.
minConfidence	xs:integer	Specifies the minimum confidence. This attribute is optional.
maxConfidence	xs:integer	Specifies the maximum confidence. This attribute is optional.

3.1.4.1.3.12 t:ConfigurationRequestDetailsType

The **ConfigurationRequestDetailsType** complex type contains a request for details of the **policy tips**. [<13>](#)

```
<xs:complexType name="ConfigurationRequestDetailsType">
  <xs:choice minOccurs="1" maxOccurs="1">
    <xs:any processContents="skip" minOccurs="0" maxOccurs="unbounded"
      namespace="##any"/>
  </xs:choice>
</xs:complexType>
```

The **ConfigurationRequestDetailsType** complex type contains an **XML schema any** element definition, as specified in [\[XMLSCHEMA1\]](#) section 3.10.2, that MUST contain an element called **PolicyNudges**. The **PolicyNudges** element can contain attributes called **OutlookVersion** and **OutlookLocale**. Values for the **OutlookVersion** attribute are ignored by the server. The **OutlookVersion** attribute can contain the client's version number. The **OutlookLocale** attribute can contain the client's culture-specific natural language code, as specified in [\[RFC4646\]](#). The **OutlookLocale** MUST be set. The response value of the **text2** element in the **localeType** complex type, as specified in section [3.1.4.1.3.23](#), is localized according to the value of the **OutlookLocale** attribute value. The **PolicyNudges** element can contain a **PolicyNudgeRules** element and a

ClassificationItems element. The **PolicyNudgeRules** element is specified in section [3.1.4.1.3.31](#). The **ClassificationItems** element contains a **ClassificationDefinitions** element, as specified in section 3.1.4.1.3.31. The **ClassificationItems** element can have an **EngineVersion** attribute. The **EngineVersion** attribute can contain the version number of the client classification library. The value of the **EngineVersion** attribute is ignored by the server.

If the **ConfigurationRequestDetailsType** complex type is empty, no **OutlookVersion** or **EngineVersion** attributes will be present, the **OutlookLocale** attribute will be set to "en-US", and the **PolicyNudgeRule** and **ClassificationDefinition** elements will not be present.

3.1.4.1.3.13 mce:DescriptionType Complex Type

The **DescriptionType** complex type specifies the description of a localized resource string. [<14>](#)

```
<xs:complexType name="DescriptionType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="default" type="xs:boolean" default="false"/>
      <xs:attribute name="langcode" type="mce:LangType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

The following table lists the attributes of the **DescriptionType** complex type.

Attribute	Type	Description
default	xs:boolean ([XMLSCHEMA2] section 3.2.2)	Specifies the rule pack identifier. This attribute has a default value of "false".
langcode	mce:LangType (section 3.1.4.1.4.2)	Specifies the language identifier, as specified in [RFC4646] . This attribute MUST be present.

3.1.4.1.3.14 mce:DetailsType Complex Type

The **DetailsType** complex type specifies the localized description of a rule pack. [<15>](#)

```
<xs:complexType name="DetailsType">
  <xs:sequence>
    <xs:element name="LocalizedDetails" type="mce:LocalizedDetailsType"
      maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="defaultLangCode" type="mce:LangType" use="required"/>
</xs:complexType>
```

The following table lists the child element of the **DetailsType** complex type.

Element	Type	Description
LocalizedDetails	mce:LocalizedDetailsType (section 3.1.4.1.3.24)	Specifies the localized information about a rule pack (section 3.1.4.1.3.52).

The following table lists the attribute of the **DetailsType** complex type.

Attribute	Type	Description
defaultLangCode	mce:LangType (section 3.1.4.1.4.2)	Specifies the language identifier, as specified in [RFC4646] , that MUST be used if the requested localized rule pack information is not available. This attribute MUST be present.

3.1.4.1.3.15 mce:EncryptionType Complex Type

The **EncryptionType** complex type specifies the encryption key and initialization vector. [<16>](#)

```
<xs:complexType name="EncryptionType">
  <xs:sequence>
    <xs:element name="Key" type="xs:normalizedString"/>
    <xs:element name="IV" type="xs:normalizedString"/>
  </xs:sequence>
</xs:complexType>
```

The following table lists the child elements of the **EncryptionType** complex type.

Element	Type	Description
Key	xs:normalizedString ([XMLSCHEMA2] section 3.3.1)	Specifies the encryption key.
IV	xs:normalizedString	Specifies the encryption initialization vector.

3.1.4.1.3.16 mce:EntityType Complex Type

The **EntityType** complex type specifies an entity classification type. [<17>](#)

```
<xs:complexType name="EntityType">
  <xs:sequence>
    <xs:element name="Pattern" type="mce:PatternType" maxOccurs="unbounded"/>
    <xs:element name="Version" type="mce:VersionedPatternType" minOccurs="0"
maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
  <xs:attribute name="patternsProximity" type="mce:ProximityType" use="required"/>
  <xs:attribute name="recommendedConfidence" type="mce:ProbabilityType"/>
  <xs:attribute name="workload" type="mce:WorkloadType"/>
</xs:complexType>
```

The following table lists the child element of the **EntityType** complex type.

Element	Type	Description
Pattern	mce:PatternType (section 3.1.4.1.3.29)	Specifies the classification rules that are matched for an entity-based classification.

Element	Type	Description
Version	mce:VersionedPatternType (section 3.1.4.1.3.65)	Specifies the version of a list of patterns.

The following table lists the attributes of the **EntityType** complex type.

Attribute	Type	Description
id	mce:GuidType (section 3.1.4.1.4.1)	Specifies the rule identifier. This attribute MUST be present.
patternsProximity	mce:ProximityType (section 3.1.4.1.4.11)	Specifies the proximity value between the primary match, as represented by the IdMatch element, as specified in section 3.1.4.1.3.29, and corroborative evidence matches, as specified by the classification engine. This attribute MUST be present.
recommendedConfidence	mce:ProbabilityType (section 3.1.4.1.4.10)	Specifies the confidence level threshold used by the classification engine to trigger an action. This attribute MUST be present.
workload	mce:WorkloadType (section 3.1.4.1.4.15)	This attribute is not used.

3.1.4.1.3.17 mce:EvidenceType Complex Type

The **EvidenceType** complex type specifies the classification rules for an affinity-based classification. [<18>](#)

```
<xs:complexType name="EvidenceType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="Match" type="mce:MatchType"/>
      <xs:element name="Any" type="mce:AnyType"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="confidenceLevel" type="mce:ProbabilityType" use="required"/>
</xs:complexType>
```

The following table lists the child elements of the **EvidenceType** complex type.

Element	Type	Description
Match	mce:MatchType (section 3.1.4.1.3.26)	Specifies a single classification rule to match. This element can occur zero or more times.
Any	mce:AnyType (section 3.1.4.1.3.4)	Specifies two or more classification rules to match. This element can occur zero or more times.

The following table lists the attribute of the **EvidenceType** complex type.

Attribute	Type	Description
confidenceLevel	mce:ProbabilityType (section 3.1.4.1.4.10)	Specifies the probability of a match. This attribute MUST be present.

3.1.4.1.3.18 m:GetServiceConfigurationResponseMessageType Complex Type

The **GetServiceConfigurationResponseMessageType** complex type contains the response message for a **GetServiceConfiguration WSDL operation**. This type extends the **ResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.65.

```
<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>
```

The following table lists the child elements of the **GetServiceConfigurationResponseMessageType** complex type.

Element	Type	Description
ResponseMessages	m:ArrayOfServiceConfigurationResponseMessageType (section 3.1.4.1.3.7)	Contains an array of service configuration response messages. This element MUST occur if there are configuration settings.

3.1.4.1.3.19 m:GetServiceConfigurationType Complex Type

The **GetServiceConfigurationType** complex type specifies the requested service configurations and identifies the sender or impersonated sender who is making the request. The impersonated sender can be different from the user account that authenticated with the server. This type extends the **m:BaseRequestType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.17.

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

```

    />
    <xs:element name="RequestedConfiguration"
      type="m:ArrayOfServiceConfigurationType"
      maxOccurs="1"
      minOccurs="1"
    />
    <xs:element minOccurs="0" maxOccurs="1" name="ConfigurationRequestDetails"
type="t:ConfigurationRequestDetailsType" />
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

The following table lists the child elements of the **GetServiceConfigurationType** complex type.

Element	Type	Description
ActingAs	t:EmailAddressType ([MS-OXWSCDATA] section 2.2.4.31)	Specifies who the caller is sending as in the case of an impersonation scenario. If this element is not present, the authenticated user is assumed to be the sender. The ActingAs element MUST be included for requesting mail tips . Otherwise, this element is optional.
RequestedConfiguration	m:ArrayOfServiceConfigurationType (section 3.1.4.1.3.8)	Specifies the requested service configurations. This element MUST be present.
ConfigurationRequestDetails	t:ConfigurationRequestDetailsType (section 3.1.4.1.3.12)	Specifies the request for DLP details. <19>

3.1.4.1.3.20 mce:GroupType Complex Type

The **GroupType** complex type specifies the terms in a keyword-based rule. [<20>](#)

```

<xs:complexType name="GroupType">
  <xs:sequence>
    <xs:choice>
      <xs:element name="Term" type="mce:TermType" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="matchStyle" default="word">
    <xs:simpleType>
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="word"/>
        <xs:enumeration value="string"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>

```

The following table lists the child element of the **GroupType** complex type.

Element	Type	Description
Term	mce:TermType (section 3.1.4.1.3.60)	Specifies a term and whether the term is case-sensitive. The term MUST be between 1 and 100 characters in length. One or more Term elements can be present.

The following table lists the attribute of the **GroupType** complex type.

Attribute	Type	Description
matchStyle	xs:simpleType ([XMLSCHEMA2] section 3.2.2)	Specifies whether the match of the rule condition and Message object is based on a word or a string. The following are the possible values for this attribute: <ul style="list-style-type: none"> word string The default value is "word". A word match is based on an entire word; a string match is based on a match of a substring of the string.

3.1.4.1.3.21 mce:IdMatchType Complex Type

The **IdMatchType** complex type specifies an entity-based primary match rule. [<21>](#)

```
<xs:complexType name="IdMatchType">
  <xs:attribute name="idRef" type="xs:string" use="required"/>
</xs:complexType>
```

The following table lists the attribute of the **IdMatchType** complex type.

Attribute	Type	Description
idRef	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies a text processor identifier.

3.1.4.1.3.22 mce:KeywordType Complex Type

The **KeywordType** complex type specifies the terms and identifier of a keyword-based rule. [<22>](#)

```
<xs:complexType name="KeywordType">
  <xs:sequence>
    <xs:element name="Group" type="mce:GroupType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:token" use="required"/>
</xs:complexType>
```

The following table lists the child element of the **KeywordType** complex type.

Element	Type	Description
Group	mce:GroupType (section 3.1.4.1.3.20)	Specifies a group of terms in a keyword-based text processor.

The following table lists the attribute of the **KeywordType** complex type.

Attribute	Type	Description
id	xs:token ([XMLSCHEMA2] section 3.3.2)	Specifies the unique identifier of a keyword-based text processor.

3.1.4.1.3.23 localeType Complex Type

The **localeType** complex type specifies the unique identifier of a keyword-based text processor. [<23>](#)

```
<xs:complexType name="localeType">
  <xs:sequence>
    <xs:element name="complianceNoteUrl" type="xs:anyURI" nillable="true" />
    <xs:element name="text1" type="xs:string" />
    <xs:element name="text2" type="xs:string" />
    <xs:element name="overrideText" type="xs:string" />
    <xs:element name="text3" type="xs:string" />
  </xs:sequence>
  <xs:attribute name="name" type="xs:language" use="required" />
</xs:complexType>
```

The following table lists the child elements of the **localeType** complex type.

Element	Type	Description
complianceNoteUrl	xs:anyURI ([XMLSCHEMA2] section 3.2.17)	Specifies the URL of a compliance note.
text1	xs:string ([XMLSCHEMA2] section 3.2.1)	This element is not used.
text2	xs:string	Specifies a policy tip message.
overrideText	xs:string	This element is not used.
text3	xs:string	This element is not used.

The following table lists the attribute of the **localeType** complex type.

Attribute	Type	Description
name	xs:language ([XMLSCHEMA2] section 3.3.3)	Specifies the language identifier for the localeType complex type.

3.1.4.1.3.24 mce:LocalizedDetailsType Complex Type

The **LocalizedDetailsType** complex type specifies the localized information about a rule pack. [<24>](#)

```

<xs:complexType name="LocalizedDetailsType">
  <xs:sequence>
    <xs:element name="PublisherName" type="mce:NameType"/>
    <xs:element name="Name" type="mce:RulePackNameType"/>
    <xs:element name="Description" type="mce:OptionalNameType"/>
  </xs:sequence>
  <xs:attribute name="langcode" type="mce:LangType" use="required"/>
</xs:complexType>

```

The following table lists the child elements of the **LocalizedDetailsType** complex type.

Element	Type	Description
PublisherName	mce:NameType (section 3.1.4.1.4.4)	Specifies the name of the rules pack publisher.
Name	mce:RulePackNameType (section 3.1.4.1.4.13)	Specifies the name of a rule pack.
Description	mce:OptionalNameType (section 3.1.4.1.4.5)	Specifies the rule pack description.

The following table lists the attribute of the **LocalizedDetailsType** complex type.

Attribute	Type	Description
langcode	mce:LangType (section 3.1.4.1.4.2)	Specifies the language identifier for the LocalizedDetailsType complex type.

3.1.4.1.3.25 mce:LocalizedStringsType Complex Type

The **LocalizedStringsType** complex type specifies localized information about classification rules for both entity-based and affinity-based rules. [<25>](#)

```

<xs:complexType name="LocalizedStringsType">
  <xs:sequence>
    <xs:element name="Resource" type="mce:ResourceType" maxOccurs="unbounded">
      <xs:key name="UniqueLangCodeUsedInNamePerResource">
        <xs:selector xpath="mce:Name"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
      <xs:key name="UniqueLangCodeUsedInDescriptionPerResource">
        <xs:selector xpath="mce:Description"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **LocalizedStringsType** complex type.

Element	Type	Description
Resource	mce:ResourceType (section 3.1.4.1.3.49)	Specifies localized information about entity-based and affinity-based based classification rules. Only one Resource element per

Element	Type	Description
		unique entity or affinity MUST exist, and the language code used in the Name and Description elements of the ResourceType complex type MUST be unique per Resource element.

3.1.4.1.3.26 mce:MatchType Complex Type

The **MatchType** complex type specifies a classification rule match. [<26>](#)

```
<xs:complexType name="MatchType">
  <xs:attribute name="idRef" type="xs:string" use="required"/>
</xs:complexType>
```

The following table lists the attribute of the **MatchType** complex type.

Attribute	Type	Description
idRef	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies a text processor identifier. This attribute MUST be present.

3.1.4.1.3.27 t:MailTipsServiceConfiguration Complex Type

The **MailTipsServiceConfiguration** complex type contains service configuration information for the **mail tips** service. This complex type extends the **ServiceConfiguration** complex type, as specified in section [3.1.4.1.3.56](#).

```
<xs:complexType name="MailTipsServiceConfiguration">
  <xs:complexContent>
    <xs:extension base="t:ServiceConfiguration">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="MailTipsEnabled"
          type="xs:boolean"/>
        <xs:element minOccurs="1" maxOccurs="1"
          name="MaxRecipientsPerGetMailTipsRequest" type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="MaxMessageSize"
          type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="LargeAudienceThreshold"
          type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="ShowExternalRecipientCount"
          type="xs:boolean"/>
        <xs:element minOccurs="1" maxOccurs="1" name="InternalDomains"
          type="t:SmtpDomainList"/>
        <xs:element minOccurs="1" maxOccurs="1" name="PolicyTipsEnabled"
          type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="LargeAudienceCap"
          type="xs:int" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the **MailTipsServiceConfiguration** complex type.

Element	Type	Description
MailTipsEnabled	xs:boolean (XMLSCHEMA2 section 3.2.2)	Specifies whether the mail tips service is available. A text value of "true" indicates that mail tips are enabled. This element MUST be present.
MaxRecipientsPerGetMailTipsRequest	xs:int (XMLSCHEMA2 section 3.3.17)	Specifies the maximum number of recipients that can be passed to the GetMailTips WSDL operation (MS-OXWMT section 3.1.4.1). This element MUST be included if the MailTipsServiceConfiguration complex type is used and MUST be a non-negative integer, as specified in XMLSCHEMA2 section 3.3.20.
MaxMessageSize	xs:int	Specifies the maximum message size that the user identified by the ActingAs element of the GetServiceConfigurationType complex type (section 3.1.4.1.3.19) can send. This element MUST be included if the MailTipsServiceConfiguration complex type is used and MUST be a non-negative integer.
LargeAudienceThreshold	xs:int	Specifies the large audience threshold for clients. This element MUST be included if the MailTipsServiceConfiguration complex type is used and MUST be a non-negative integer.
ShowExternalRecipientCount	xs:boolean	Specifies whether clients that are using the GetMailTips WSDL operation have to show mail tips that indicate the number of external recipients to which a message is addressed. A text value of "true" indicates that clients have to show mail tips. This element MUST be included if the MailTipsServiceConfiguration complex type is used.
InternalDomains	t:SmtpDomainList (section 3.1.4.1.3.59)	Specifies a list of Simple Mail Transfer Protocol (SMTP) domains that are internal to an organization. This information is provided so that mail client software can provide external recipient mail tips when offline. This element MUST be included if the MailTipsServiceConfiguration complex type is used.
PolicyTipsEnabled	xs:boolean	Specifies whether policy tips are enabled. <27> A text value of "true" indicates that policy tips are enabled.
LargeAudienceCap	xs:int	Specifies the maximum number of recipients for a mailbox item. <28>

3.1.4.1.3.28 orType Complex Type

The **orType** complex type specifies a logical **OR** for rule predicates. This type extends the **predicateElementMultipleChildType** complex type, as specified in section [3.1.4.1.3.35.<29>](#)

```
<xs:complexType name="orType">
  <xs:complexContent>
    <xs:extension base="predicateElementMultipleChildType">
      <xs:attribute name="earlyOut" use="optional">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <!-- Case insensitive: true, yes, y, 1, false, no, n, 0-->
            <xs:pattern
              value="([Tt][Rr][Uu][Ee])|([Yy]([Ee][Ss])?)|([1])|
                ([Ff][Aa][Ll][Ss][Ee])|([Nn][Oo]?)|([0])" />
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the attribute of the **orType** complex type

Attribute	Type	Description
earlyOut	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies whether the logical OR acts as an early out operator. This attribute is optional. The following are the valid values for this attribute: <ul style="list-style-type: none"> ▪ true ▪ yes ▪ y ▪ 1 ▪ false ▪ no ▪ n ▪ 0 The early out operator can be used to capture additional information after the OR operator evaluates to true .

3.1.4.1.3.29 mce:PatternType Complex Type

The **PatternType** complex type specifies the pattern for an entity classification. [<30>](#)

```
<xs:complexType name="PatternType">
  <xs:sequence>
    <xs:element name="IdMatch" type="mce:IdMatchType"/>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element name="Match" type="mce:MatchType"/>
      <xs:element name="Any" type="mce:AnyType"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="confidenceLevel" type="mce:ProbabilityType" use="required"/>
</xs:complexType>
```



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

The following table lists the child elements of the **PatternType** complex type.

Element	Type	Description
IdMatch	mce:IdMatchType (section 3.1.4.1.3.21)	Specifies an identifier of a text processor that is used as the primary match for this entity-based classification rule. This element MUST be present.
Match	mce:MatchType (section 3.1.4.1.3.26)	Specifies a corroborative evidence match.
Any	mce:AnyType (section 3.1.4.1.3.4)	Specifies one or more corroborative evidence matches where if any of the corroborative evidences are matched, then the rule condition is processed as a match.

The following table lists the attribute of the **PatternType** complex type.

Attribute	Type	Description
confidenceLevel	mce:ProbabilityType (section 3.1.4.1.4.10)	Specifies the confidence level for this entity when the pattern is matched by the classification engine. This attribute is required.

3.1.4.1.3.30 t:PolicyNudgeRulesServiceConfiguration Complex Type

The **PolicyNudgeRulesServiceConfiguration** complex type contains the DLP configuration data. The top-level element defined by the **any** element definition is the **PolicyNudgeRulesConfiguration** element, as specified in section [3.1.4.1.2.3.<31>](#)

```
<xs:complexType name="PolicyNudgeRulesServiceConfiguration">
  <xs:sequence>
    <xs:any processContents="skip" minOccurs="0" maxOccurs="unbounded"
      namespace="##any"/>
  </xs:sequence>
</xs:complexType>
```

3.1.4.1.3.31 PolicyNudgeRulesConfigurationType Complex Type

The **PolicyNudgeRulesConfigurationType** complex type specifies the set of DLP rules and classification definitions that are sent to a client. [<32>](#)

```
<xs:complexType name="PolicyNudgeRulesConfigurationType">
  <xs:sequence>
    <xs:element name="PolicyNudgeRules" type="PolicyNudgeRulesType">
      <!-- Ensure that 2 DLP rules (rule) don't have the same name -->
      <xs:unique name="ruleNameUnique">
        <xs:selector xpath="PolicyNudgeRule/rule" />
        <xs:field xpath="@name"/>
      </xs:unique>
      <!-- Ensure that 2 DLP rule envelopes (PolicyNudgeRule) don't have
```

```

    the same id -->
    <xs:unique name="policyNudgeRuleIDUnique">
      <xs:selector xpath="PolicyNudgeRule" />
      <xs:field xpath="@id"/>
    </xs:unique>
  </xs:element>
  <xs:element name="ClassificationDefinitions"
    type="ClassificationDefinitionsType">
    <!-- Ensure that 2 Classification Definition envelopes
      (ClassificationDefinition) don't have the same id -->
    <xs:unique name="classificationDefinitionIDUnique">
      <xs:selector xpath="ClassificationDefinition" />
      <xs:field xpath="@id"/>
    </xs:unique>
  </xs:element>
</xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **PolicyNudgeRulesConfigurationType** complex type.

Element	Type	Description
PolicyNudgeRules	mce:PolicyNudgeRulesType (section 3.1.4.1.3.32)	Specifies a collection of DLP rules. Each DLP rule MUST have a unique name and identifier.
ClassificationDefinitions	ClassificationDefinitionsType (section 3.1.4.1.3.9)	Specifies a collection of classification definitions. Each classification definition MUST have a unique identifier.

3.1.4.1.3.32 mce:PolicyNudgeRulesType Complex Type

The **PolicyNudgeRulesType** complex type specifies a collection of DLP rules. This complex type extends the **ApplyType** complex type, as specified in section [3.1.4.1.3.5.<33>](#)

```

<xs:complexType name="PolicyNudgeRulesType">
  <xs:complexContent>
    <xs:extension base="ApplyType">
      <xs:sequence>
        <xs:element name="PolicyNudgeRule" type="PolicyNudgeRuleType"
          minOccurs="0" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The following table lists the child element of the **PolicyNudgeRulesType** complex type.

Element	Type	Description
PolicyNudgeRule	PolicyNudgeRuleType (section 3.1.4.1.3.33)	Specifies a single DLP rule.

3.1.4.1.3.33 PolicyNudgeRuleType Complex Type

The **PolicyNudgeRuleType** complex type specifies a single DLP rule. <34>

```
<xs:complexType name="PolicyNudgeRuleType">
  <xs:sequence>
    <xs:element name="rule" type="ruleType" />
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required" />
  <xs:attribute name="version" type="xs:long" use="required" />
  <xs:anyAttribute processContents="skip" />
</xs:complexType>
```

The following table lists the child element of the **PolicyNudgeRuleType** complex type.

Element	Type	Description
rule	ruleType (section 3.1.4.1.3.53)	Specifies a rule definition.

The following table lists the attributes of the **PolicyNudgeRuleType** complex type.

Attribute	Type	Description
id	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies a DLP rule identifier. This attribute MUST be present.
version	xs:long ([XMLSCHEMA2] section 3.3.16)	Specifies a DLP rule version. This attribute MUST be present.
	xs:anyAttribute ([XMLSCHEMA1] section 3.10)	Indicates that the server can send non-predefined attributes in a response to the client. The client MUST send back the same attributes in subsequent requests.

3.1.4.1.3.34 predicateElementConstantType Complex Type

The **predicateElementConstantType** complex type specifies the type of all constant rule predicates. All elements of this type **MUST** be empty. <35>

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

3.1.4.1.3.35 predicateElementMultipleChildType Complex Type

The **predicateElementMultipleChildType** complex type specifies multiple predicates and logical operators for a classification definition. <36>

```
<xs:complexType name="predicateElementMultipleChildType">
  <xs:choice minOccurs="1" maxOccurs="unbounded">
    <xs:element name="and" type="predicateElementMultipleChildType" />
    <xs:element name="classification" type="classificationType" />
    <xs:element name="classifications" type="predicateElementSingleChildType" />
    <xs:element name="false" type="predicateElementConstantType" />
    <xs:element name="not" type="predicateElementSingleChildType" />
    <xs:element name="or" type="orType" />
    <xs:element name="recipient" type="recipientType" />
    <xs:element name="recipients" type="predicateElementSingleChildType" />
  </xs:choice>
</xs:complexType>
```

```

    <xs:element name="sender" type="senderType" />
    <xs:element name="senders" type="predicateElementSingleChildType" />
    <xs:element name="true" type="predicateElementConstantType" />
  </xs:choice>
</xs:complexType>

```

The following table lists the elements of the **predicateElementMultipleChildType** complex type.

Element	Type	Description
and	predicateElementMultipleChildType	Specifies a logical AND for one or more predicates.
classification	classificationType (section 3.1.4.1.3.11)	Specifies a classification identifier.
classifications	predicateElementSingleChildType (section 3.1.4.1.3.36)	This element is not used.
false	predicateElementConstantType (section 3.1.4.1.3.34)	Specifies the false constant predicate.
not	predicateElementSingleChildType	Specifies a not predicate.
or	orType (section 3.1.4.1.3.28)	Specifies a logical OR .
recipient	recipientType (section 3.1.4.1.3.46)	Specifies a message recipient that can be used in a predicate.
recipients	predicateElementSingleChildType	This element is not used.
sender	senderType (section 3.1.4.1.3.55)	Specifies a message sender that can be used in a predicate.
senders	predicateElementSingleChildType	This element is not used.
true	predicateElementConstantType	Specifies the true constant predicate.

3.1.4.1.3.36 predicateElementSingleChildType Complex Type

The **predicateElementSingleChildType** complex type specifies a single predicate or logical operator for a classification definition. [<37>](#)

```

<xs:complexType name="predicateElementSingleChildType">
  <xs:choice>
    <xs:element name="and" type="predicateElementMultipleChildType" />
    <xs:element name="classification" type="classificationType" />
    <xs:element name="classifications" type="predicateElementSingleChildType" />
    <xs:element name="false" type="predicateElementConstantType" />
    <xs:element name="not" type="predicateElementSingleChildType" />
    <xs:element name="or" type="orType" />
    <xs:element name="recipient" type="recipientType" />
    <xs:element name="recipients" type="predicateElementSingleChildType" />
    <xs:element name="sender" type="senderType" />
    <xs:element name="senders" type="predicateElementSingleChildType" />
    <xs:element name="true" type="predicateElementConstantType" />
  </xs:choice>
</xs:complexType>

```

The following table describes the child elements of the **predicateElementSingleChildType** complex type.

Element	Type	Description
and	predicateElementMultipleChildType (section 3.1.4.1.3.35)	Specifies a logical AND for one or more predicates.
classification	classificationType (section 3.1.4.1.3.11)	Specifies a classification identifier.
classifications	predicateElementSingleChildType	Specifies a container for many classification identifiers.
false	predicateElementConstantType (section 3.1.4.1.3.34)	Specifies the false constant predicate.
not	predicateElementSingleChildType	Specifies a not predicate.
or	orType (section 3.1.4.1.3.28)	Specifies a logical OR .
recipient	recipientType (section 3.1.4.1.3.46)	Specifies a message recipient that can be used in a predicate.
recipients	predicateElementSingleChildType	Specifies a container for many message recipients.
sender	senderType (section 3.1.4.1.3.55)	Specifies a message sender that can be used in a predicate.
senders	predicateElementSingleChildType	Specifies a container for many message senders.
true	predicateElementConstantType	Specifies the true constant predicate.

3.1.4.1.3.37 t:ProtectionRuleActionType Complex Type

The **ProtectionRuleActionType** complex type specifies the action that the client can take if the condition part of the associated rule evaluates to "true".

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

The following table lists the child element of the **ProtectionRuleActionType** complex type.

Element	Type	Description
Argument	t:ProtectionRuleArgumentType (section 3.1.4.1.3.39)	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

Element	Type	Description
		RightsProtectMessage action name (section 3.1.4.1.4.6) is the only supported value for the Name attribute, and it MUST have a single argument.

The following table lists the attribute of the **ProtectionRuleActionType** complex type.

Attribute	Type	Description
Name	t:ProtectionRuleActionKindType (section 3.1.4.1.4.6)	Specifies the name of the action. This attribute MUST be present.

3.1.4.1.3.38 t:ProtectionRuleAndType Complex Type

The **ProtectionRuleAndType** complex type specifies that there has to 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>
```

The following table lists the child elements of the **ProtectionRuleAndType** complex type.

Element	Type	Description
AllInternal	t:ProtectionRuleAllInternalType (section 3.1.4.1.4.7)	Evaluates to "true" if all recipients of an email 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 (section 3.1.4.1.3.41)	Evaluates to "true" if any recipient of the email message matches any of the specified recipients in the child Value elements.
SenderDepartments	t:ProtectionRuleSenderDepartmentsType (section 3.1.4.1.3.42)	Evaluates to "true" if the department of the sender matches any specified department in the child Value elements.
True	t:ProtectionRuleTrueType (section 3.1.4.1.4.8)	Specifies a condition that evaluates to "true".

3.1.4.1.3.39 t:ProtectionRuleArgumentType Complex Type

The **ProtectionRuleArgumentType** complex 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>
```

The following table lists the attributes of the **ProtectionRuleArgumentType** complex type.

Attribute	Type	Description
Value	xs:string (XMLSCHEMA2 section 3.2.1)	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.

3.1.4.1.3.40 t:ProtectionRuleConditionType Complex Type

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

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

The following table lists the child elements of the **ProtectionRuleConditionType** complex type.

Element	Type	Description
AllInternal	t:ProtectionRuleAllInternalType (section 3.1.4.1.4.7)	Evaluates to "true" if all recipients of an email message are internal to the sender's organization. If this element exists, the And , RecipientIs , SenderDepartments , and True elements MUST NOT be direct child nodes of elements of type ProtectionRuleConditionType .
And	t:ProtectionRuleAndType (section 3.1.4.1.3.38)	Evaluates to "true" if all child conditions evaluate to "true". There MUST be more than one protection rule child condition. If this element exists, the AllInternal ,

Element	Type	Description
		RecipientIs , SenderDepartments , and True elements MUST NOT be direct child nodes of elements of type ProtectionRuleConditionType .
RecipientIs	t:ProtectionRuleRecipientIsType (section 3.1.4.1.3.41)	Evaluates to "true" if any recipient of the email message matches any of the specified recipients in the child Value elements. If this element exists, the And , AllInternal , SenderDepartments , and True elements MUST NOT be direct child nodes of elements of type ProtectionRuleConditionType .
SenderDepartments	t:ProtectionRuleSenderDepartmentsType (section 3.1.4.1.3.42)	Evaluates to "true" if the department of the sender matches any of the specified departments in the child Value elements. If this element exists, the And , RecipientIs , AllInternal , and True elements MUST NOT be direct child nodes of elements of type ProtectionRuleConditionType .
True	t:ProtectionRuleTrueType (section 3.1.4.1.4.8)	Specifies a condition that evaluates to "true". If this element exists, the And , RecipientIs , SenderDepartments , and AllInternal elements MUST NOT be direct child nodes of elements of type ProtectionRuleConditionType .

3.1.4.1.3.41 t:ProtectionRuleRecipientIsType Complex Type

The **ProtectionRuleRecipientIsType** complex type specifies the **RecipientIs** predicate. The semantics of the **RecipientIs** predicate is a condition that evaluates to "true" if any **recipients** of the email 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>
```

The following table lists the child element of the **ProtectionRuleRecipientIsType** complex type.

Element	Type	Description
Value	t:ProtectionRuleValueType (section 3.1.4.1.4.9)	Specifies an argument to the RecipientIs condition. This element MUST occur at least once.

3.1.4.1.3.42 t:ProtectionRuleSenderDepartmentsType Complex Type

The **ProtectionRuleSenderDepartmentsType** complex type specifies the **SenderDepartments** predicate. The semantics of the **SenderDepartments** predicate is a condition that evaluates to "true" if the department of the sender of the email 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>
```

The following table lists the child element of the **ProtectionRuleSenderDepartmentsType** complex type.

Element	Type	Description
Value	t:ProtectionRuleValueType (section 3.1.4.1.4.9)	Specifies an argument to the SenderDepartments condition. This element MUST occur at least once.

3.1.4.1.3.43 t:ProtectionRulesServiceConfiguration Complex Type

The **ProtectionRulesServiceConfiguration** complex type specifies the configuration of the protection rules service. The configuration comprises a list of rules, a list of the internal **domains**, and a refresh interval. This type extends the **t:ServiceConfiguration** complex type, as specified in section [3.1.4.1.3.56](#).

```
<xs:complexType name="ProtectionRulesServiceConfiguration">
  <xs:complexContent>
    <xs:extension base="t:ServiceConfiguration">
      <xs:sequence>
        <xs:element name="Rules" type="t:ArrayOfProtectionRulesType"
          maxOccurs="1" minOccurs="1" />
        <xs:element name="InternalDomains" type="t:SmtDomainList"
          maxOccurs="1" minOccurs="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>
```

The following table lists the child elements of the **ProtectionRulesServiceConfiguration** complex type.

Element	Type	Description
Rules	t:ArrayOfProtectionRulesType	Specifies the collection of rules to be evaluated.

Element	Type	Description
	(section 3.1.4.1.3.6)	This element MUST be included if the ProtectionRulesServiceConfiguration complex type is used.
InternalDomains	t:SmtpDomainList (section 3.1.4.1.3.59)	Specifies the list of internal SMTP domains of the organization. This element MUST be included if the ProtectionRulesServiceConfiguration complex type is used.

The following table lists the attribute of the **ProtectionRulesServiceConfiguration** complex type.

Attribute	Type	Description
RefreshInterval	1. xs:int ([XMLSCHEMA2] section 3.3.17)	Specifies how often, in whole hours, the client SHOULD request protection rules from the server. This attribute MUST be included if the ProtectionRulesServiceConfiguration complex type is used.

3.1.4.1.3.44 t:ProtectionRuleType Complex Type

The **ProtectionRuleType** complex 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>
```

The following table lists the child elements of the **ProtectionRuleType** complex type.

Element	Type	Definition
Condition	t:ProtectionRuleConditionType (section 3.1.4.1.3.40)	Specifies the condition to be satisfied for the action part of the rule to be executed. This element MUST be included if the ProtectionRuleType complex type is used.

Element	Type	Definition
Action	t:ProtectionRuleActionType (section 3.1.4.1.3.37)	Specifies the action to be executed if the condition part of the rule matches. This element MUST be included if the ProtectionRuleType complex type is used.

The following table lists the attributes of the **ProtectionRuleType** complex type.

Attribute	Type	Definition
Name	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies the name of the rule. This attribute MUST be included if the ProtectionRuleType complex type is used. This attribute value MUST contain a string of at least one character.
UserOverridable	xs:boolean ([XMLSCHEMA2] section 3.2.2)	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 complex type is used.
Priority	xs:int ([XMLSCHEMA2] section 3.3.17)	Specifies the rule priority. The lower bound MUST be 1, which specifies the highest priority. This attribute MUST be included if the ProtectionRuleType complex type is used. This attribute value MUST contain an integer value of at least 1.

3.1.4.1.3.45 mce:PublisherType Complex Type

The **PublisherType** complex type specifies the unique identifier of a rule pack publisher. [<38>](#)

```
<xs:complexType name="PublisherType">
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
</xs:complexType>
```

The following table lists the attribute of the **PublisherType** complex type.

Attribute	Type	Definition
id	mce:GuidType (section 3.1.4.1.4.1)	Specifies the rules pack publisher's identifier. This attribute MUST be present.

3.1.4.1.3.46 recipientType Complex Type

The **recipientType** complex type specifies a **recipient** in a classification definition. No more than one of the attributes in this type can be present. [<39>](#)

```
<xs:complexType name="recipientType">
  <xs:attribute name="address" type="xs:string" use="optional" />
  <xs:attribute name="domain" type="xs:string" use="optional" />
  <xs:attribute name="scope" use="optional">
    <xs:simpleType>
      <xs:restriction base="xs:string">
```

```

        <xs:enumeration value="Internal"/>
        <xs:enumeration value="External"/>
        <xs:enumeration value="ExternalNonPartner"/>
        <xs:enumeration value="ExternalPartner"/>
    </xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>

```

The following table lists the attributes of the **recipientType** complex type.

Attribute	Type	Description
address	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies a recipient's email address .
domain	xs:string	Specifies a recipient's domain .
scope	xs:string	Specifies a recipient's organizational scope as compared to the target mailbox user. The applicable values for this attribute are as follows: <ul style="list-style-type: none"> ▪ Internal ▪ External ▪ ExternalNonPartner ▪ ExternalPartner

3.1.4.1.3.47 mce:RegexType Complex Type

The **RegexType** complex type specifies a regular expression text processor. [<40>](#)

```

<xs:complexType name="RegexType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="id" type="xs:token" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

```

The following table lists the attribute of the **RegexType** complex type.

Attribute	Type	Description
id	xs:token ([XMLSCHEMA2] section 3.3.2)	Specifies the regular expression text processor identifier.

3.1.4.1.3.48 mce:ResourceNameType Complex Type

The **ResourceNameType** complex type specifies a localized name for a classification rule. [<41>](#)

```

<xs:complexType name="ResourceNameType">
  <xs:simpleContent>

```

```

<xs:extension base="xs:string">
  <xs:attribute name="default" type="xs:boolean" default="false"/>
  <xs:attribute name="langcode" type="mce:LangType" use="required"/>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

```

The following table lists the attributes of the **ResourceNameType** complex type.

Attribute	Type	Description
default	xs:boolean (XMLSCHEMA2 section 3.2.1)	Specifies whether this is the default resource name. A text value of "true" indicates that this is the default resource name. The default value is "false".
langcode	mce:LangType (section 3.1.4.1.4.2)	Specifies the language code that represents the localized name of a classification rule. The langcode attribute value of all Name elements (section 3.1.4.1.3.49) MUST be unique per Resource element (section 3.1.4.1.3.25). This attribute MUST be present.

3.1.4.1.3.49 mce:ResourceType Complex Type

The **ResourceType** complex type specifies the localized names and descriptions of a classification rule. [<42>](#)

```

<xs:complexType name="ResourceType">
  <xs:sequence>
    <xs:element name="Name" type="mce:ResourceNameType" maxOccurs="unbounded"/>
    <xs:element name="Description" type="mce:DescriptionType" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="idRef" type="mce:GuidType" use="required"/>
</xs:complexType>

```

The following table lists the child elements of the **ResourceType** complex type.

Element	Type	Description
Name	mce:ResourceNameType (section 3.1.4.1.3.48)	Specifies a localized resource name.
Description	mce:DescriptionType (section 3.1.4.1.3.13)	Specifies a localized resource description. A Resource element, as specified in section 3.1.4.1.3.25 , of type ResourceType MUST have only one Description element with a value default attribute set to the value of true .

The following table lists the attribute of the **ResourceType** complex type.

Attribute	Type	Description
idRef	mce:GuidType (section 3.1.4.1.4.1)	Specifies the classification rule identifier that this resource represents.

3.1.4.1.3.50 mce:RulePackageContainerType Complex Type

The **RulePackageContainerType** complex type contains a single rule package. [<43>](#)

```
<xs:complexType name="RulePackageContainerType">
  <xs:sequence>
    <xs:element name="RulePackage" type="mce:RulePackageType" />
  </xs:sequence>
</xs:complexType>
```

The following table lists the child element of the **RulePackageContainerType** complex type.

Element	Type	Description
RulePackage	mce:RulePackageType (section 3.1.4.1.3.51)	Specifies a classification rule package. This element MUST be present.

3.1.4.1.3.51 mce:RulePackageType Complex Type

The **RulePackageType** complex type specifies the rule pack identifier and the set of classification definitions rules. [<44>](#) A rule pack is a collection of **rules** and associated resources.

```
<xs:complexType name="RulePackageType">
  <xs:sequence>
    <xs:element name="RulePack" type="mce:RulePackType"/>
    <xs:element name="Rules" type="mce:RulesType">
      <xs:key name="UniqueRuleId">
        <xs:selector
          xpath="mce:Entity|mce:Affinity|mce:Version/mce:Entity|mce:Version/mce:Affinity"/>
        <xs:field xpath="@id"/>
      </xs:key>
      <xs:key name="UniqueProcessorId">
        <xs:selector xpath="mce:Regex|mce:Keyword|mce:Fingerprint"/></xs:selector>
        <xs:field xpath="@id"/>
      </xs:key>
      <xs:key name="UniqueResourceIdRef">
        <xs:selector xpath="mce:LocalizedStrings/mce:Resource"/>
        <xs:field xpath="@idRef"/>
      </xs:key>
      <xs:keyref name="ReferencedRuleMustExist" refer="mce:UniqueRuleId">
        <xs:selector xpath="mce:LocalizedStrings/mce:Resource"/>
        <xs:field xpath="@idRef"/>
      </xs:keyref>
      <xs:keyref name="RuleMustHaveResource" refer="mce:UniqueResourceIdRef">
        <xs:selector
          xpath="mce:Entity|mce:Affinity|mce:Version/mce:Entity|mce:Version/mce:Affinity"/>
        <xs:field xpath="@id"/>
      </xs:keyref>
    </xs:element>
  </xs:sequence>
```

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

The following table lists the child elements of the **RulePackageType** complex type.

Element	Type	Description
RulePack	mce:RulePackType (section 3.1.4.1.3.52)	Specifies the version, publisher information, and localized information of a rule pack.
Rules	mce:RulesType (section 3.1.4.1.3.54)	Specifies a collection of classification rules, text processors, and localized information of the classification rules.

3.1.4.1.3.52 mce:RulePackType Complex Type

The **RulePackType** complex type specifies the version, identifier, and publisher of a rule pack. [<45>](#)

```
<xs:complexType name="RulePackType">
  <xs:sequence>
    <xs:element name="Version" type="mce:VersionType"/>
    <xs:element name="Publisher" type="mce:PublisherType"/>
    <xs:element name="Details" type="mce:DetailsType">
      <xs:key name="UniqueLangCodeInLocalizedDetails">
        <xs:selector xpath="mce:LocalizedDetails"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
      <xs:keyref name="DefaultLangCodeMustExist"
        refer="mce:UniqueLangCodeInLocalizedDetails">
        <xs:selector xpath="."/>
        <xs:field xpath="@defaultLangCode"/>
      </xs:keyref>
    </xs:element>
    <xs:element name="Encryption" type="mce:EncryptionType" minOccurs="0"
      maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
</xs:complexType>
```

The following table describes the child elements of the **RulePackType** complex type.

Element	Type	Description
Version	mce:VersionType (section 3.1.4.1.3.62)	Specifies the rule pack version. This element MUST be present.
Publisher	mce:PublisherType (section 3.1.4.1.3.45)	Specifies the rule pack publisher. This element MUST be present.
Details	mce:DetailsType (section 3.1.4.1.3.14)	Specifies the rule pack localized information. This element MUST be present.
Encryption	mce:EncryptionType (section 3.1.4.1.3.15)	Specifies the rule pack encryption key and initialization vector. This MUST be present only when the rule pack is encrypted.

The following table describes the attribute of the **RulePackType** complex type.

Attribute	Type	Description
id	mce:GuidType (section 3.1.4.1.4.1)	Specifies the rule pack identifier.

3.1.4.1.3.53 ruleType Complex Type

The **ruleType** complex type specifies a DLP rule. [<46>](#)

```
<xs:complexType name="ruleType">
  <xs:sequence>
    <xs:element name="version" type="versionType" minOccurs="1" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required" />
  <xs:attribute name="id" type="xs:string" use="required" />
  <xs:attribute name="description" type="xs:string" use="required" />
</xs:complexType>
```

The following table describes the child element of the **ruleType** complex type.

Element	Type	Description
version	versionType (section 3.1.4.1.3.63)	Specifies the minimum client version that can use this rule and the rule conditions and actions. This element MUST be present.

The following table describes the attributes of the **ruleType** complex type.

Attribute	Type	Description
name	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies the rule name. This attribute MUST be present.
id	xs:string	Specifies the rule identifier. This attribute MUST be present.
description	xs:string	Specifies the rule description. This attribute MUST be present.

3.1.4.1.3.54 mce:RulesType Complex Type

The **RulesType** complex type specifies the rules collection in a rules package. [<47>](#)

```
<xs:complexType name="RulesType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="Entity" type="mce:EntityType"/>
      <xs:element name="Affinity" type="mce:AffinityType"/>
      <xs:element name="Version" type="mce:VersionedRuleType"/>
    </xs:choice>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element name="Regex" type="mce:RegexType"/>
      <xs:element name="Keyword" type="mce:KeywordType"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>
```



```

    <xs:element name="Fingerprint" type="mce:FingerprintType"/>
  </xs:choice>
  <xs:element name="LocalizedStrings" type="mce:LocalizedStringsType"/>
</xs:sequence>
</xs:complexType>

```

The following table lists the child elements of the **RulesType** complex type.

Element	Type	Description
Entity	mce:EntityType (section 3.1.4.1.3.16)	Specifies an entity-based rules classification definition.
Affinity	mce:AffinityType (section 3.1.4.1.3.3)	Specifies an affinity-based rules classification definition.
Regex	mce:RegexType (section 3.1.4.1.3.47)	Specifies a regular expression text processor.
Keyword	mce:KeywordType (section 3.1.4.1.3.22)	Specifies a keyword text processor.
LocalizedStrings	mce:LocalizedStringsType (section 3.1.4.1.3.25)	Specifies the localized information for an entity or affinity classification rule.
Version	mce:VersionedRuleType	Specifies version of rules classification definition.
Fingerprint	mce:FingerprintType	Specifies a fingerprint text processor.

3.1.4.1.3.55 senderType Complex Type

The **senderType** complex type specifies a sender in a classification definition. [<48>](#)

```

<xs:complexType name="senderType">
  <xs:attribute name="address" type="xs:string" use="optional" />
  <xs:attribute name="domain" type="xs:string" use="optional" />
</xs:complexType>

```

The following table describes the attributes of the **senderType** complex type.

Attribute	Type	Description
address	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies a sender's email address .
domain	xs:string	Specifies a sender's domain .

3.1.4.1.3.56 t:ServiceConfiguration Complex Type

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

```

<xs:complexType name="ServiceConfiguration">

```

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

3.1.4.1.3.57 m:ServiceConfigurationResponseMessageType Complex Type

The **ServiceConfigurationResponseMessageType** complex type specifies service configuration settings. This complex type extends the **ResponseMessageType** complex type, as specified in [\[MS-OXWSCDATA\]](#) section 2.2.4.65.

```
<xs:complexType name="ServiceConfigurationResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="MailTipsConfiguration"
          type="t:MailTipsServiceConfiguration" minOccurs="0"
          maxOccurs="1" />
        <xs:element name="UnifiedMessagingConfiguration"
          type="t:UnifiedMessageServiceConfiguration"
          maxOccurs="1" minOccurs="0" />
        <xs:element name="ProtectionRulesConfiguration"
          type="t:ProtectionRulesServiceConfiguration"
          maxOccurs="1" minOccurs="0" />
        <xs:element name="PolicyNudgeRulesConfiguration"
          type="t:PolicyNudgeRulesServiceConfiguration" minOccurs="0"
          maxOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

The following table lists the child elements of the **ServiceConfigurationResponseMessageType** complex type.

Element	Type	Description
MailTipsConfiguration	t:MailTipsServiceConfiguration (section 3.1.4.1.3.27)	Contains service configuration information for the mail tips service. This element MUST occur for mail tips service configuration.
UnifiedMessagingConfiguration	t:UnifiedMessageServiceConfiguration (section 3.1.4.1.3.61)	Contains service configuration information for the Unified Messaging service. This element MUST occur for Unified Messaging service configuration.
ProtectionRulesConfiguration	t:ProtectionRulesServiceConfiguration (section 3.1.4.1.3.43)	Contains service configuration information for the protection rules service. This element MUST occur for protection rules service configuration.
PolicyNudgeRulesConfiguration	t:PolicyNudgeRulesServiceConfiguration (section 3.1.4.1.3.30)	Contains DLP rules and classification definitions. <49>

3.1.4.1.3.58 t:SmtpDomain Complex Type

The **SmtpDomain** complex 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>
```

The following table lists the attributes of the **SmtpDomain** complex type.

Attribute	Type	Definition
Name	xs:string (XMLSCHEMA2) section 3.2.1)	Specifies the name of a domain. This attribute MUST be set.
IncludeSubdomains	xs:boolean (XMLSCHEMA2) section 3.2.2)	Specifies whether subdomains of the domain identified by the Name attribute are included. A text value of "true" indicates that subdomains are included. This attribute is optional. The default value is "false".

3.1.4.1.3.59 t:SmtpDomainList Complex Type

The **SmtpDomainList** complex 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>
```

The following table lists the child element of the **SmtpDomainList** complex type.

Element	Type	Definition
Domain	t:SmtpDomain (section 3.1.4.1.3.58)	Specifies a single SMTP domain. This element can occur zero or more times. It occurs once for each domain in the user's organization. If there are no internal domains, this element does not occur.

3.1.4.1.3.60 mce:TermType Complex Type

The **TermType** complex type specifies a term and whether the term is case-sensitive. [<50>](#) This type extends the **RestrictedTermType** simple type, as specified in section [3.1.4.1.4.12](#).

```
<xs:complexType name="TermType">
  <xs:simpleContent>
    <xs:extension base="mce:RestrictedTermType">
      <xs:attribute name="caseSensitive" type="xs:boolean" default="false"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

```

</xs:simpleContent>
</xs:complexType>

```

The following table describes the attribute of the **TermType** complex type.

Attribute	Type	Description
caseSensitive	xs:boolean ([XMLSCHEMA2] section 3.2.2)	Specifies whether the term comparison is case sensitive. A value of "true" indicates that the comparison is case sensitive. The default value is "false".

3.1.4.1.3.61 t:UnifiedMessageServiceConfiguration Complex Type

The **UnifiedMessageServiceConfiguration** complex type specifies the configuration for the **Unified Messaging** service. This type extends the **ServiceConfiguration** complex type, as specified in section [3.1.4.1.3.56](#).

```

<xs:complexType name="UnifiedMessageServiceConfiguration">
  <xs:complexContent>
    <xs:extension base="t:ServiceConfiguration">
      <xs:sequence>
        <xs:element name="UmEnabled" type="xs:boolean"
          maxOccurs="1" minOccurs="1" />
        <xs:element name="PlayOnPhoneDialString" type="xs:string"
          maxOccurs="1" minOccurs="1" />
        <xs:element name="PlayOnPhoneEnabled" type="xs:boolean"
          maxOccurs="1" minOccurs="1" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

The following table lists the child elements of the **UnifiedMessageServiceConfiguration** complex type.

Element	Type	Description
UmEnabled	xs:boolean ([XMLSCHEMA2] section 3.2.2)	Specifies whether Unified Messaging is enabled. A text value of "true" indicates that Unified Messaging is enabled. This element MUST be included if the UnifiedMessageServiceConfiguration complex type is used.
PlayOnPhoneDialString	xs:string ([XMLSCHEMA2] section 3.2.1)	Specifies the telephone number for play-on-phone. This element MUST be included if the UnifiedMessageServiceConfiguration complex type is used.
PlayOnPhoneEnabled	xs:boolean	Specifies whether play-on-phone is enabled. A text value of "true" indicates that play-on-phone is enabled. This element MUST be included if the UnifiedMessageServiceConfiguration complex type is used.

3.1.4.1.3.62 VersionType Complex Type

The **VersionType** complex type specifies the rule pack version.

```
<xs:complexType name="VersionType">
  <xs:attribute name="major" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="minor" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="build" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="revision" type="xs:unsignedShort" use="required"/>
</xs:complexType>
```

The following table lists the attributes of the **VersionType** complex type.

Attribute	Type	Description
major	xs:unsignedShort ([XMLSCHEMA2] section 3.3.23)	Specifies the rule pack major version number.
minor	xs:unsignedShort	Specifies the rule pack minor version number.
build	xs:unsignedShort	Specifies the rule pack build number.
revision	xs:unsignedShort	Specifies the rule pack revision number.

3.1.4.1.3.63 versionType Complex Type

The **versionType** complex type specifies the minimum client version that can use the rule and the rule conditions and actions. [<51>](#)

```
<xs:complexType name="versionType">
  <xs:sequence>
    <xs:element name="condition" type="predicateElementSingleChildType" />
    <xs:element name="actions" type="actionsType" />
  </xs:sequence>
  <xs:attribute name="minRequiredVersion" type="minRequiredVersionType"
    use="required" />
</xs:complexType>
```

The following table describes the child elements of the **versionType** complex type.

Element	Type	Description
condition	predicateElementSingleChildType (section 3.1.4.1.3.36)	Specifies the rule conditions.
actions	actionsType (section 3.1.4.1.3.1)	Specifies the rule actions to take if the rule conditions are met.

The following table describes the attribute of the **versionType** complex type.

Attribute	Type	Description
minRequiredVersion	minRequiredVersionType (section 3.1.4.1.4.3)	Specifies the minimum required client version that can use this rule.

3.1.4.1.3.64 mce:VersionedRuleType Complex Type

The **VersionedRuleType** complex type specifies the required classification engine version for a list of data classification rules. [<52>](#)

```
<xs:complexType name="VersionedRuleType">
  <xs:choice maxOccurs="unbounded">
    <xs:element name="Entity" type="mce:EntityType"/>
    <xs:element name="Affinity" type="mce:AffinityType"/>
  </xs:choice>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
```

The following table describes the child elements of the **VersionedRuleType** complex type.

Element	Type	Description
Entity	mce:EntityType (section 3.1.4.1.3.16)	Specifies an entity-based rules classification definition.
Affinity	mce:AffinityType (section 3.1.4.1.3.3)	Specifies an affinity-based rules classification definition.

The following table lists the attribute of the **VersionedRuleType** complex type.

Attribute	Type	Description
minEngineVersion	mce:EngineVersionType (section 3.1.4.1.4.17)	Specifies the required engine version of classification rules.

3.1.4.1.3.65 mce:VersionedPatternType Complex Type

The **VersionedPatternType** complex type specifies the required classification engine version for a list of pattern elements. [<53>](#)

```
<xs:complexType name="VersionedPatternType">
  <xs:sequence>
    <xs:element name="Pattern" type="mce:PatternType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
```

The following table lists the child elements of the **VersionedPatternType** complex type.

Element	Type	Description
Pattern	mce:PatternType (section 3.1.4.1.3.29)	Specifies the classification rules that are matched for an entity-based classification.

The following table lists the attribute of the **VersionedPatternType** complex type.

Attribute	Type	Description
minEngineVersion	mce:EngineVersionType (section 3.1.4.1.4.17)	Specifies the required engine version of classification pattern elements.

3.1.4.1.3.66 mce:VersionedEvidenceType Type

The **VersionedEvidenceType** complex type specifies the required classification engine version for a list of evidence elements. [<54>](#)

```
<xs:complexType name="VersionedEvidenceType">
  <xs:sequence>
    <xs:element name="Evidence" type="mce:EvidenceType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
```

The following table lists the child elements of the **VersionedEvidenceType** complex type.

Element	Type	Description
Evidence	mce:EvidenceType (section 3.1.4.1.3.17)	Specifies the rules that are matched for an affinity-based classification.

The following table lists the attribute of the **VersionedEvidenceType** complex type.

Attribute	Type	Description
minEngineVersion	mce:EngineVersionType (section 3.1.4.1.4.17)	Specifies the required engine version of evidence elements.

3.1.4.1.3.67 mce:FingerprintType Type

The **FingerprintType** complex type specifies properties of the fingerprint. [<55>](#)

```
<xs:complexType name="FingerprintType">
  <xs:simpleContent>
    <xs:extension base="mce:FingerprintValueType">
      <xs:attribute name="id" type="xs:token" use="required"/>
      <xs:attribute name="threshold" type="mce:ProbabilityType" use="required"/>
      <xs:attribute name="shingleCount" type="xs:positiveInteger" use="required"/>
      <xs:attribute name="description" type="xs:string" use="optional"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

```

    </xs:simpleContent>
</xs:complexType>

```

The following table lists the attribute of the **FingerprintType** complex type.

Attribute	Type	Description
id	xs:token	Specifies the identifier of this fingerprint.
threshold	mce:ProbabilityType (section 3.1.4.1.4.10)	Specifies the percentage of containment that the fingerprint text processor will identify it as a matching item. The threshold level has to be a value between 1 and 100.
shingleCount	xs:positiveInteger	Specifies the number of shingles of the document that is used to generate fingerprint.
description	xs:string	Specifies the description of the fingerprint.

3.1.4.1.4 Simple Types

The following table summarizes the **XML schema** simple type definitions that are specific to this operation.

Simple type	Description
GuidType (section 3.1.4.1.4.1)	Specifies the regular expression pattern of a GUID .
LangType (section 3.1.4.1.4.2)	Specifies the natural language identifiers as specified in [RFC4646] .
minRequiredVersionType (section 3.1.4.1.4.3)	Specifies the format of the minimum client version that can implement the rule.
NameType (section 3.1.4.1.4.4)	Specifies a string format that has a minimum length of one character; has a maximum length of 256 characters; and does not contain a carriage return, a line feed, or a tab character.
OptionalNameType (section 3.1.4.1.4.5)	Specifies a string format that has no minimum length; has a maximum length of 256 characters; and does not contain a carriage return, a line feed, or a tab character.
ProtectionRuleActionKindType (section 3.1.4.1.4.6)	Specifies the actions that are supported by the protection rules
ProtectionRuleAllInternalType (section 3.1.4.1.4.7)	Specifies the AllInternal predicate.

ProtectionRuleTrueType (section 3.1.4.1.4.8)	Specifies the True predicate.
ProtectionRuleValueType (section 3.1.4.1.4.9)	Specifies additional arguments to the RecipientIs (section 3.1.4.1.3.41) and SenderDepartments (section 3.1.4.1.3.42) predicates.
ProbabilityType (section 3.1.4.1.4.10)	Specifies the degree of confidence that the server rule processing engines have identified a matching item.
ProximityType (section 3.1.4.1.4.11)	Specifies the threshold for how far apart the primary match and the corroborative evidences can be in order for the entity or affinity to be considered a match.
RestrictedTermType (section 3.1.4.1.4.12)	Specifies the terms for keyword-based processing. The minimum length can be one character; the maximum length MUST be 100 characters.
RulePackNameType (section 3.1.4.1.4.13)	Specifies the format of a rule pack name. A rule pack name MUST have no fewer than one character and no more than 64 characters. A rule pack name MUST NOT contain a carriage return, a line feed, a tab character, leading spaces, trailing spaces, or sequences of two or more spaces.
ServiceConfigurationType (section 3.1.4.1.4.14)	Specifies the service configurations that are returned in the response.
WorkloadType (section 3.1.4.1.4.15)	Not used.
yesnoType (section 3.1.4.1.4.16)	Specifies whether override or false positive settings are enabled.
EngineVersionType (section 3.1.4.1.4.17)	Specifies the supported engine version for rules, entity-based or affinity-based classifications.
FingerprintValueType (section 3.1.4.1.4.18)	Specifies the fingerprint value for a rule.

3.1.4.1.4.1 mce:GuidType Simple Type

The **GuidType** simple type specifies the regular expression pattern of a **GUID**.[<56>](#)

```
<xs:simpleType name="GuidType" final="#all">
  <xs:restriction base="xs:token">
    <xs:pattern value="[0-9a-fA-F]{8}\-([0-9a-fA-F]{4}\-){3}[0-9a-fA-F]{12}"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.4.2 mce:LangType Simple Type

The **LangType** simple type specifies the natural language identifiers, as specified in [\[RFC4646\]](#).[<57>](#)

```

<xs:simpleType name="LangType">
  <xs:union memberTypes="xs:language">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value=""/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

```

3.1.4.1.4.3 minRequiredVersionType Simple Type

The **minRequiredVersionType** simple type specifies the format of the minimum client version that can implement the rule. Attributes of type **minRequiredVersionType** take the form of 4 16-bit integers separated by periods. The first integer MUST be present. The last three integers SHOULD be present. [<58>](#)

```

<xs:simpleType name="minRequiredVersionType">
  <xs:restriction base="xs:string">
    <xs:pattern value="[0-6]?[0-9]{1,4}(\.[0-6]?[0-9]{1,4}){0,3}"/>
  </xs:restriction>
</xs:simpleType>

```

3.1.4.1.4.4 mce:NameType Simple Type

The **NameType** simple type specifies a string format that has a minimum length of one character; has a maximum length of 256 characters; and does not contain a carriage return, a line feed, or a tab character. [<59>](#)

```

<xs:simpleType name="NameType">
  <xs:restriction base="xs:normalizedString">
    <xs:minLength value="1"/>
    <xs:maxLength value="256"/>
  </xs:restriction>
</xs:simpleType>

```

3.1.4.1.4.5 mce:OptionalNameType Simple Type

The **OptionalNameType** simple type specifies a string format that has no minimum length; has a maximum length of 256 characters; and does not contain a carriage return, a line feed, or a tab character. [<60>](#)

```

<xs:simpleType name="OptionalNameType">
  <xs:restriction base="xs:normalizedString">
    <xs:minLength value="0"/>
    <xs:maxLength value="256"/>
  </xs:restriction>
</xs:simpleType>

```

3.1.4.1.4.6 t:ProtectionRuleActionKindType Simple Type

The **ProtectionRuleActionKindType** simple type specifies the actions that are supported by the protection rules.

```

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

```

```
</xs:restriction>
</xs:simpleType>
```

The following table lists the value that is defined by the **ProtectionRuleActionKindType** simple type.

Value	Meaning
RightsProtectMessage	The specific permission template to apply to the message.

Only the **RightsProtectMessage** action is supported. The value MUST be **RightsProtectMessage**. The protection rules are used to apply a specific set of permissions.

3.1.4.1.4.7 t:ProtectionRuleAllInternalType Simple Type

The **ProtectionRuleAllInternalType** simple type specifies the **AllInternal** predicate. The semantics of the **AllInternal** predicate is that the predicate evaluates to "true" if all **recipients** of the email message are internal to the organization of the sender of that email message.

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

3.1.4.1.4.8 t:ProtectionRuleTrueType Simple Type

The **ProtectionRuleTrueType** simple type specifies the **True** predicate. The semantics of the **True** predicate is that the condition evaluates to "true".

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

3.1.4.1.4.9 t:ProtectionRuleValueType Simple Type

The **ProtectionRuleValueType** simple type specifies additional arguments to the **RecipientIs** predicate of the **ProtectionRuleRecipientIsType** complex type, as specified in section [3.1.4.1.3.41](#), and the **SenderDepartments** predicate of the **ProtectionRuleSenderDepartmentsType** complex type, as specified in section [3.1.4.1.3.42](#). 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>
```

3.1.4.1.4.10 mce:ProbabilityType Simple Type

The **ProbabilityType** simple type specifies the degree of confidence that the server rule processing engines have identified a matching item. The confidence level has to be a value between 1 and 100. [<61>](#)

```
<xs:simpleType name="ProbabilityType">
  <xs:restriction base="xs:integer">
    <xs:minInclusive value="1"/>
    <xs:maxInclusive value="100"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.4.11 mce:ProximityType Simple Type

The **ProximityType** simple type specifies the threshold for how far apart the primary match and the corroborative evidences can be in order for the entity or affinity to be considered a match. [<62>](#)

```
<xs:simpleType name="ProximityType">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.4.12 mce:RestrictedTermType Simple Type

The **RestrictedTermType** simple type specifies the terms for keyword-based processing. The minimum length can be one character; the maximum length MUST be 100 characters. [<63>](#)

```
<xs:simpleType name="RestrictedTermType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="100"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.4.13 mce:RulePackNameType Simple Type

The **RulePackNameType** simple type specifies the format of a rule pack name. A rule pack name MUST have no fewer than one character and no more than 64 characters. A rule pack name MUST NOT contain a carriage return, a line feed, a tab character, leading spaces, trailing spaces, or sequences of two or more spaces. [<64>](#)

```
<xs:simpleType name="RulePackNameType">
  <xs:restriction base="xs:token">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.4.14 t:ServiceConfigurationType Simple Type

The **ServiceConfigurationType** simple type 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="MailTips" />
    <xs:enumeration value="UnifiedMessagingConfiguration" />
    <xs:enumeration value="ProtectionRules" />
    <xs:enumeration value="PolicyNudges" />
  </xs:restriction>
</xs:simpleType>
</xs:list>
</xs:simpleType>

```

The following table lists the values that are defined by the **ServiceConfigurationType** simple type.

Value	Meaning
MailTips	Represents the mail tips service configuration.
UnifiedMessagingConfiguration	Represents the Unified Messaging service configuration.
ProtectionRules	Represents the protection rules service configuration.
PolicyNudges	Represents the DLP service configuration. <65>

3.1.4.1.4.15 mce:WorkloadType Simple Type

The **WorkloadType** simple type is not used. [<66>](#)

```

<xs:simpleType name="WorkloadType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Exchange"/>
    <xs:enumeration value="Outlook"/>
  </xs:restriction>
</xs:simpleType>

```

3.1.4.1.4.16 yesnoType Simple Type

The **yesnoType** simple type specifies whether override or false positive settings are enabled. [<67>](#)

```

<xs:simpleType name="yesnoType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="yes" />
    <xs:enumeration value="no" />
  </xs:restriction>
</xs:simpleType>

```

The following table lists the values that are defined by the **yesnoType** simple type.

Value	Description
yes	Indicates that the setting is enabled.
no	Indicates that the setting is disabled.

3.1.4.1.4.17 mce:EngineVersionType Simple Type

The **EngineVersionType** simple type specifies the supported engine version for rules, including entity-based or affinity-based classification rules. [<68>](#)

```
<xs:simpleType name="EngineVersionType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="15.0.780.0"/>
    <xs:enumeration value="15.0.847.013"/>
  </xs:restriction>
</xs:simpleType>
```

The following table lists the values that are defined by the EngineVersionType simple type.

Value	Description
15.0.780.9	All rule, pattern or evidence elements that reference the fingerprint SHOULD use this version.
15.0.847.013	Specifies the second version of the engine version. <69>

3.1.4.1.4.18 mce:FingerprintValueType Simple Type

The FingerprintValueType simple type specifies the fingerprint value constraints. It must be a based64 encoded string with 2732 characters. [<70>](#)

```
<xs:simpleType name="FingerprintValueType">
  <xs:restriction base="xs:string">
    <xs:minLength value="2732"/>
    <xs:maxLength value="2732"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.4.1.5 Attributes

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

3.1.4.1.6 Groups

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

3.1.4.1.7 Attribute Groups

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

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

4.1 GetServiceConfiguration Operation Request

The following example shows how to get **mail tips** for a user by using the **GetServiceConfiguration** operation, as described in section [3.1.4.1](#). The request is for the mail tips configuration for user1.

```
<?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 Operation Response

The following **XML** shows a successful response from the **GetServiceConfiguration** service request in section 4.2.

```
<?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:MaxRecipientsPerGetMailTipsRequest>
    <t:MaxMessageSize
        xmlns:t="http://schemas.microsoft.com/exchange/services/
        2006/types">
        10485760
    </t:MaxMessageSize>
    <t:LargeAudienceThreshold
        xmlns:t="http://schemas.microsoft.com/exchange/services/
        2006/types">
        25
    </t:LargeAudienceThreshold>
    <t>ShowExternalRecipientCount
        xmlns:t="http://schemas.microsoft.com/exchange/services/
        2006/types">
        false
    </t>ShowExternalRecipientCount>
    <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 GetServiceConfiguration Operation Response

4.3.1 SOAP Exception

The following **XML** shows a **SOAP fault** caused by the failure of request schema validation. The **GetServiceConfiguration** operation request, as described in section [3.1.4.1](#), was called. The request failed because it did not include a required element in the **GetServiceConfigurationType** complex type, as described in section [3.1.4.1.3.19](#).

```

<?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 Operation Error Response

The following example shows the error response that occurs when the user specified in the **ActingAs** element of the **GetServiceConfigurationType** complex type, as described in section [3.1.4.1.3.19](#), is not 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

None.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

The XML files that are listed in the following table are required in order to implement the functionality specified in this document.

File name	Description	Section
MS-OXWCONFIG.wsdl	Contains the WSDL for the implementation of this protocol.	6
MS-OXWCONFIG-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	7.6
MS-OXWCONFIG-types.xsd	Contains the XML schema type definitions that are used in this protocol.	7.9

These files have to be placed in a common folder in order for the WSDL to validate and operate. Also, any schema files that are included in or imported into the MS-OXWCONFIG-types.xsd or MS-OXWCONFIG-messages.xsd schemas have to be placed in the common folder with these files.

This section contains the contents of the MS-OXWCONFIG.wsdl file.

```
<?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"
  targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
  xmlns:wsd="http://schemas.xmlsoap.org/wsdl/"
  xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified"
      version="Exchange2016"
      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">
      <xs:include schemaLocation="MS-OXWCONFIG-messages.xsd"/>
    </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/GetServiceConfiguration"/>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

```
<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: Full XML Schema

For ease of implementation, the following sections provide the full **XML schema** for this protocol.

Schema name	Prefix	Section
Messages schema	m:	7.6
Types schema	t:	7.9

These files have to be placed in a common folder in order for the **WSDL** to validate and operate. Also, any schema files that are included in or imported into the MS-OXWCONFIG-types.xsd or MS-OXWCONFIG-messages.xsd schemas have to be placed in the common folder along with the files listed in the table.

7.1 Classification Rule Package Container Type Schema

This section contains the contents of the ClassificationRulePackageContainerSchemaType.xsd file and information about additional files that this schema file requires to operate correctly. [<71>](#)

ClassificationRulePackageContainerSchemaType.xsd includes and imports the file listed in the following table. This file has to be placed in a common folder in order for the schema to validate and operate. Also, any schema files that are included in or imported into the ClassificationRulePackageContainerSchemaType.xsd schema have to be placed in the common folder along with the file listed in the table.

File name	Defining section
RulePackageTypes.xsd	7.8

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:mce="http://schemas.microsoft.com/office/2011/mce"
  targetNamespace="http://schemas.microsoft.com/office/2011/mce"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:include schemaLocation="RulePackageTypes.xsd" />
  <xs:complexType name="RulePackageContainerType">
    <xs:sequence>
      <xs:element name="RulePackage" type="mce:RulePackageType" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

7.2 DLP Rule Schema

This section contains the contents of the PolicyNudgeRuleSchema.xsd file and information about additional files that this schema file requires to operate correctly. [<72>](#)

PolicyNudgeRuleSchema.xsd includes the file listed in the following table. This file has to be placed in a common folder in order for the schema to validate and operate. Also, any schema files that are included in or imported into the PolicyNudgeRuleSchema.xsd schemas have to be placed in the common folder along with the files listed in the table.

File name	Defining section
PolicyNudgeRuleSchemaTypes.xsd	7.3

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<xs:schema
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="PolicyNudgeRuleSchemaTypes.xsd" />
  <xs:element name="rule" type="ruleType" />
</xs:schema>

```

7.3 DLP Rule Types Schema

This section contains the contents of the PolicyNudgeRuleSchemaTypes.xsd file. [<73>](#)

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="ruleType">
    <xs:sequence>
      <xs:element name="version" type="versionType" minOccurs="1"
        maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required" />
    <xs:attribute name="id" type="xs:string" use="required" />
    <xs:attribute name="description" type="xs:string" use="required" />
  </xs:complexType>

  <xs:complexType name="versionType">
    <xs:sequence>
      <xs:element name="condition" type="predicateElementSingleChildType" />
      <xs:element name="actions" type="actionsType" />
    </xs:sequence>
    <xs:attribute name="minRequiredVersion" type="minRequiredVersionType"
      use="required" />
  </xs:complexType>

  <xs:simpleType name="minRequiredVersionType">
    <xs:restriction base="xs:string">
      <xs:pattern value="[0-6]?[0-9]{1,4}(\.[0-6]?[0-9]{1,4}){0,3}" />
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="predicateElementSingleChildType">
    <xs:choice>
      <xs:element name="and" type="predicateElementMultipleChildType" />
      <xs:element name="classification" type="classificationType" />
      <xs:element name="classifications" type="predicateElementSingleChildType" />
      <xs:element name="false" type="predicateElementConstantType" />
      <xs:element name="not" type="predicateElementSingleChildType" />
      <xs:element name="or" type="orType" />
      <xs:element name="recipient" type="recipientType" />
      <xs:element name="recipients" type="predicateElementSingleChildType" />
      <xs:element name="sender" type="senderType" />
      <xs:element name="senders" type="predicateElementSingleChildType" />
      <xs:element name="true" type="predicateElementConstantType" />
    </xs:choice>
  </xs:complexType>

  <xs:complexType name="predicateElementMultipleChildType">
    <xs:choice minOccurs="1" maxOccurs="unbounded">
      <xs:element name="and" type="predicateElementMultipleChildType" />
      <xs:element name="classification" type="classificationType" />
      <xs:element name="classifications" type="predicateElementSingleChildType" />
      <xs:element name="false" type="predicateElementConstantType" />
      <xs:element name="not" type="predicateElementSingleChildType" />
      <xs:element name="or" type="orType" />
    </xs:choice>
  </xs:complexType>

```

```

        <xs:element name="recipient" type="recipientType" />
        <xs:element name="recipients" type="predicateElementSingleChildType" />
        <xs:element name="sender" type="senderType" />
        <xs:element name="senders" type="predicateElementSingleChildType" />
        <xs:element name="true" type="predicateElementConstantType" />
    </xs:choice>
</xs:complexType>

<xs:complexType name="predicateElementConstantType" />

<xs:complexType name="classificationType">
    <xs:attribute name="rulePackId" type="xs:string" use="required" />
    <xs:attribute name="id" type="xs:string" use="required" />
    <xs:attribute name="minCount" type="xs:integer" use="optional" />
    <xs:attribute name="maxCount" type="xs:integer" use="optional" />
    <xs:attribute name="minConfidence" type="xs:integer" use="optional" />
    <xs:attribute name="maxConfidence" type="xs:integer" use="optional" />
</xs:complexType>

<xs:complexType name="orType">
    <xs:complexContent>
        <xs:extension base="predicateElementMultipleChildType">
            <xs:attribute name="earlyOut" use="optional">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:pattern
value="([Tt][Rr][Uu][Ee])|([Yy]([Ee][Ss])?)|([1])|([Ff][Aa][Ll][Ss][Ee])|([Nn][Oo]?)|([0])"
/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="recipientType">
    <xs:attribute name="address" type="xs:string" use="optional" />
    <xs:attribute name="domain" type="xs:string" use="optional" />
    <xs:attribute name="scope" use="optional">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Internal"/>
                <xs:enumeration value="External"/>
                <xs:enumeration value="ExternalNonPartner"/>
                <xs:enumeration value="ExternalPartner"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>

<xs:complexType name="senderType">
    <xs:attribute name="address" type="xs:string" use="optional" />
    <xs:attribute name="domain" type="xs:string" use="optional" />
</xs:complexType>

<xs:complexType name="actionsType">
    <xs:choice maxOccurs="unbounded">
        <xs:element name="block" type="actionTypeType" />
        <xs:element name="notify" type="actionTypeType" />
    </xs:choice>
</xs:complexType>

<xs:complexType name="actionTypeType">
    <xs:sequence>
        <xs:element name="message">
            <xs:complexType>
                <xs:sequence minOccurs="1" maxOccurs="unbounded">
                    <xs:element name="locale" type="localeType" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>

```

```

    </xs:complexType>
  </xs:element>

  <xs:element name="override">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="justification">
          <xs:complexType>
            <xs:attribute name="type" use="required">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:enumeration value="none" />
                  <xs:enumeration value="optional" />
                  <xs:enumeration value="required" />
                </xs:restriction>
              </xs:simpleType>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="allow" type="yesnoType" use="required" />
    </xs:complexType>
  </xs:element>
  <xs:element name="falsePositive">
    <xs:complexType>
      <xs:attribute name="allow" type="yesnoType" use="required" />
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="localeType">
  <xs:sequence>
    <xs:element name="complianceNoteUrl" type="xs:anyURI" nillable="true" />
    <xs:element name="text1" type="xs:string" />
    <xs:element name="text2" type="xs:string" />
    <xs:element name="overrideText" type="xs:string" />
    <xs:element name="text3" type="xs:string" />
  </xs:sequence>
  <xs:attribute name="name" type="xs:language" use="required" />
</xs:complexType>

<xs:simpleType name="yesnoType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="yes" />
    <xs:enumeration value="no" />
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

7.4 DLP Rules Configuration Schema

This section contains the contents of the PolicyNudgeRulesConfigurationSchema.xsd file and information about additional files that this schema file requires to operate correctly. [<74>](#)

PolicyNudgeRulesConfigurationSchema.xsd includes the file listed in the following table. This file has to be placed in a common folder in order for the schema to validate and operate. Also, any schema files that are included in or imported into the PolicyNudgeRulesConfigurationSchema.xsd have to be placed in the common folder along with the files listed in the table.

File name	Defining section
PolicyNudgeRulesConfigurationSchemaTypes.xsd	7.5


```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="PolicyNudgeRulesConfigurationSchemaTypes.xsd" />
  <xs:element name="PolicyNudgeRulesConfiguration" type="PolicyNudgeRulesConfigurationType"
  />
</xs:schema>

```

7.5 DLP Rules Configuration Types Schema

This section contains the contents of the PolicyNudgeRulesConfigurationSchemaTypes.xsd file and information about additional files that this schema file requires to operate correctly. [<75>](#)

PolicyNudgeRulesConfigurationSchemaTypes.xsd includes or imports the files listed in the following table. To operate correctly, these files have to be present in the folder that contains the PolicyNudgeRulesConfigurationSchemaTypes file.

File name	Defining section
PolicyNudgeRuleSchemaTypes.xsd	7.3
ClassificationRulePackageContainerSchemaType.xsd	7.1

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:mce="http://schemas.microsoft.com/office/2011/mce"
  >
  <xs:include schemaLocation="PolicyNudgeRuleSchemaTypes.xsd" />
  <xs:import namespace="http://schemas.microsoft.com/office/2011/mce"
    schemaLocation="ClassificationRulePackageContainerSchemaType.xsd" />

  <xs:complexType name="PolicyNudgeRulesConfigurationType">
    <xs:sequence>
      <xs:element name="PolicyNudgeRules" type="PolicyNudgeRulesType">
        <xs:unique name="ruleNameUnique">
          <xs:selector xpath="PolicyNudgeRule/rule" />
          <xs:field xpath="@name"/>
        </xs:unique>
        <xs:unique name="policyNudgeRuleIDUnique">
          <xs:selector xpath="PolicyNudgeRule" />
          <xs:field xpath="@id"/>
        </xs:unique>
      </xs:element>
      <xs:element name="ClassificationDefinitions"
        type="ClassificationDefinitionsType">
        <xs:unique name="classificationDefinitionIDUnique">
          <xs:selector xpath="ClassificationDefinition" />
          <xs:field xpath="@id"/>
        </xs:unique>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="ApplyType">
    <xs:attribute name="apply" use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="([Tt][Rr][Uu][Ee])|([Ff][Aa][Ll][Ss][Ee])" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>

```

```

</xs:complexType>

<xs:complexType name="PolicyNudgeRulesType">
  <xs:complexContent>
    <xs:extension base="ApplyType">
      <xs:sequence>
        <xs:element name="PolicyNudgeRule" type="PolicyNudgeRuleType"
          minOccurs="0" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="PolicyNudgeRuleType">
  <xs:sequence>
    <xs:element name="rule" type="ruleType" />
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required" />
  <xs:attribute name="version" type="xs:long" use="required" />
  <xs:anyAttribute processContents="skip" />
</xs:complexType>

<xs:complexType name="ClassificationDefinitionsType">
  <xs:complexContent>
    <xs:extension base="ApplyType">
      <xs:sequence>
        <xs:element name="ClassificationDefinition"
          type="ClassificationDefinitionType" minOccurs="0"
          maxOccurs="unbounded" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="ClassificationDefinitionType">
  <xs:complexContent>
    <xs:extension base="mce:RulePackageContainerType">
      <xs:attribute name="id" type="xs:string" use="required" />
      <xs:attribute name="version" type="xs:long" use="required" />
      <xs:anyAttribute processContents="skip" />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
</xs:schema>

```

7.6 Messages Schema

This section contains the contents of the MS-OXWCONFIG-messages.xsd file and information about additional files that this schema file requires to operate correctly.

MS-OXWCONFIG-messages.xsd includes and imports the files listed in the following table. These files have to be placed in a common folder in order for the schema to validate and operate. Also, any schema files that are included in or imported into the MS-OXWCONFIG-messages.xsd schema have to be placed in the common folder along with the files listed in the table.

File name	Defining section/specification
MS-OXWSCDATA-messages.xsd	[MS-OXWSCDATA] section 7.1
MS-OXWCONFIG-types.xsd	7.9

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="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"
    elementFormDefault="qualified" version="Exchange2016" id="messages">
<xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
    schemaLocation="MS-OXWCONFIG-types.xsd"/>
<xs:include schemaLocation="MS-OXWSCDATA-messages.xsd"/>
<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="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:element minOccurs="0" maxOccurs="1"
          name="ConfigurationRequestDetails"
          type="t:ConfigurationRequestDetailsType" />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="ServiceConfigurationResponseMessageType">
  <xs:complexContent>
    <xs:extension base="m:ResponseMessageType">
      <xs:sequence>
        <xs:element name="MailTipsConfiguration"
          type="t:MailTipsServiceConfiguration" minOccurs="0" maxOccurs="1"/>
        <xs:element name="UnifiedMessagingConfiguration"
          type="t:UnifiedMessageServiceConfiguration" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="ProtectionRulesConfiguration"
          type="t:ProtectionRulesServiceConfiguration" minOccurs="0" maxOccurs="1"/>
        <xs:element name="PolicyNudgeRulesConfiguration"
          type="t:PolicyNudgeRulesServiceConfiguration" 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>

```

7.7 Rule Package Schema

This section contains the contents of the RulePackage.xsd file and information about additional files that this schema file requires to operate correctly. [76](#)

RulePackage.xsd includes the file listed in the following table. To operate correctly, these files have to be present in the folder that contains the RulePackage.xsd file.

File name	Defining section
RulePackageTypes.xsd	7.8

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:mce="http://schemas.microsoft.com/office/2011/mce"
  targetNamespace="http://schemas.microsoft.com/office/2011/mce"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  id="RulePackageSchema">
  <xs:include schemaLocation="RulePackageTypes.xsd"/>
  <xs:element name="RulePackage" type="mce:RulePackageType"/>
</xs:schema>
```

7.8 Rule Package Types Schema

This section contains the contents of the RulePackageTypes.xsd file. [77](#)

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:mce="http://schemas.microsoft.com/office/2011/mce"
  targetNamespace="http://schemas.microsoft.com/office/2011/mce"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  id="RulePackageSchema">
  <xs:simpleType name="LangType">
    <xs:union memberTypes="xs:language">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value=""/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
  <xs:simpleType name="GuidType" final="#all">
    <xs:restriction base="xs:token">
      <xs:pattern value="[0-9a-fA-F]{8}\-([0-9a-fA-F]{4}\-){3}[0-9a-fA-F]{12}"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="RulePackageType">
    <xs:sequence>
      <xs:element name="RulePack" type="mce:RulePackType"/>
      <xs:element name="Rules" type="mce:RulesType">
        <xs:key name="UniqueRuleId">
          <xs:selector
            xpath="mce:Entity|mce:Affinity|mce:Version/mce:Entity|mce:Version/mce:Affinity"/>
          <xs:field xpath="@id"/>
        </xs:key>
        <xs:key name="UniqueProcessorId">
          <xs:selector xpath="mce:Regex|mce:Keyword|mce:Fingerprint"/></xs:selector>
          <xs:field xpath="@id"/>
        </xs:key>
        <xs:key name="UniqueResourceIdRef">
          <xs:selector xpath="mce:LocalizedStrings/mce:Resource"/>
          <xs:field xpath="@idRef"/>
        </xs:key>
      </xs:sequence>
    </xs:complexType>
  </xs:schema>
```

```

    <xs:keyref name="ReferencedRuleMustExist" refer="mce:UniqueRuleId">
      <xs:selector xpath="mce:LocalizedStrings/mce:Resource"/>
      <xs:field xpath="@idRef"/>
    </xs:keyref>
    <xs:keyref name="RuleMustHaveResource" refer="mce:UniqueResourceIdRef">
      <xs:selector
xpath="mce:Entity|mce:Affinity|mce:Version|mce:Entity|mce:Version|mce:Affinity"/>
      <xs:field xpath="@id"/>
    </xs:keyref>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="RulePackType">
  <xs:sequence>
    <xs:element name="Version" type="mce:VersionType"/>
    <xs:element name="Publisher" type="mce:PublisherType"/>
    <xs:element name="Details" type="mce:DetailsType">
      <xs:key name="UniqueLangCodeInLocalizedDetails">
        <xs:selector xpath="mce:LocalizedDetails"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
      <xs:keyref name="DefaultLangCodeMustExist"
refer="mce:UniqueLangCodeInLocalizedDetails">
        <xs:selector xpath="."/>
        <xs:field xpath="@defaultLangCode"/>
      </xs:keyref>
    </xs:element>
    <xs:element name="Encryption" type="mce:EncryptionType" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
</xs:complexType>
<xs:complexType name="VersionType">
  <xs:attribute name="major" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="minor" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="build" type="xs:unsignedShort" use="required"/>
  <xs:attribute name="revision" type="xs:unsignedShort" use="required"/>
</xs:complexType>
<xs:complexType name="PublisherType">
  <xs:attribute name="id" type="mce:GuidType" use="required"/>
</xs:complexType>
<xs:complexType name="LocalizedDetailsType">
  <xs:sequence>
    <xs:element name="PublisherName" type="mce:NameType"/>
    <xs:element name="Name" type="mce:RulePackNameType"/>
    <xs:element name="Description" type="mce:OptionalNameType"/>
  </xs:sequence>
  <xs:attribute name="langcode" type="mce:LangType" use="required"/>
</xs:complexType>
<xs:complexType name="DetailsType">
  <xs:sequence>
    <xs:element name="LocalizedDetails" type="mce:LocalizedDetailsType"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="defaultLangCode" type="mce:LangType" use="required"/>
</xs:complexType>
<xs:complexType name="EncryptionType">
  <xs:sequence>
    <xs:element name="Key" type="xs:normalizedString"/>
    <xs:element name="IV" type="xs:normalizedString"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="RulePackNameType">
  <xs:restriction base="xs:token">
    <xs:minLength value="1"/>
    <xs:maxLength value="64"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NameType">
  <xs:restriction base="xs:normalizedString">

```

```

        <xs:minLength value="1"/>
        <xs:maxLength value="256"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="OptionalNameType">
    <xs:restriction base="xs:normalizedString">
        <xs:minLength value="0"/>
        <xs:maxLength value="256"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="RestrictedTermType">
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="100"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="RulesType">
    <xs:sequence>
        <xs:choice maxOccurs="unbounded">
            <xs:element name="Entity" type="mce:EntityType"/>
            <xs:element name="Affinity" type="mce:AffinityType"/>
            <xs:element name="Version" type="mce:VersionedRuleType"/>
        </xs:choice>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="Regex" type="mce:RegexType"/>
            <xs:element name="Keyword" type="mce:KeywordType"/>
            <xs:element name="Fingerprint" type="mce:FingerprintType"/>
        </xs:choice>
        <xs:element name="LocalizedStrings" type="mce:LocalizedStringsType"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="EntityType">
    <xs:sequence>
        <xs:element name="Pattern" type="mce:PatternType" maxOccurs="unbounded"/>
        <xs:element name="Version" type="mce:VersionedPatternType" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="id" type="mce:GuidType" use="required"/>
    <xs:attribute name="patternsProximity" type="mce:ProximityType" use="required"/>
    <xs:attribute name="recommendedConfidence" type="mce:ProbabilityType"/>
    <xs:attribute name="workload" type="mce:WorkloadType"/>
</xs:complexType>
<xs:complexType name="PatternType">
    <xs:sequence>
        <xs:element name="IdMatch" type="mce:IdMatchType"/>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="Match" type="mce:MatchType"/>
            <xs:element name="Any" type="mce:AnyType"/>
        </xs:choice>
    </xs:sequence>
    <xs:attribute name="confidenceLevel" type="mce:ProbabilityType" use="required"/>
</xs:complexType>
<xs:complexType name="AffinityType">
    <xs:sequence>
        <xs:element name="Evidence" type="mce:EvidenceType" maxOccurs="unbounded"/>
        <xs:element name="Version" type="mce:VersionedEvidenceType" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="id" type="mce:GuidType" use="required"/>
    <xs:attribute name="evidencesProximity" type="mce:ProximityType" use="required"/>
    <xs:attribute name="thresholdConfidenceLevel" type="mce:ProbabilityType" use="required"/>
    <xs:attribute name="workload" type="mce:WorkloadType"/>
</xs:complexType>
<xs:complexType name="EvidenceType">
    <xs:sequence>
        <xs:choice maxOccurs="unbounded">
            <xs:element name="Match" type="mce:MatchType"/>
            <xs:element name="Any" type="mce:AnyType"/>
        </xs:choice>

```

```

    </xs:sequence>
    <xs:attribute name="confidenceLevel" type="mce:ProbabilityType" use="required"/>
</xs:complexType>
<xs:complexType name="IdMatchType">
  <xs:attribute name="idRef" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="MatchType">
  <xs:attribute name="idRef" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="AnyType">
  <xs:sequence>
    <xs:choice maxOccurs="unbounded">
      <xs:element name="Match" type="mce:MatchType"/>
      <xs:element name="Any" type="mce:AnyType"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="minMatches" type="xs:nonNegativeInteger" default="1"/>
  <xs:attribute name="maxMatches" type="xs:nonNegativeInteger" use="optional"/>
</xs:complexType>
<xs:simpleType name="ProximityType">
  <xs:restriction base="xs:positiveInteger">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProbabilityType">
  <xs:restriction base="xs:integer">
    <xs:minInclusive value="1"/>
    <xs:maxInclusive value="100"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="WorkloadType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Exchange"/>
    <xs:enumeration value="Outlook"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="EngineVersionType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="15.0.780.0"/>
    <xs:enumeration value="15.0.847.013"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="VersionedRuleType">
  <xs:choice maxOccurs="unbounded">
    <xs:element name="Entity" type="mce:EntityType"/>
    <xs:element name="Affinity" type="mce:AffinityType"/>
  </xs:choice>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
<xs:complexType name="VersionedPatternType">
  <xs:sequence>
    <xs:element name="Pattern" type="mce:PatternType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
<xs:complexType name="VersionedEvidenceType">
  <xs:sequence>
    <xs:element name="Evidence" type="mce:EvidenceType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="minEngineVersion" type="mce:EngineVersionType" use="required" />
</xs:complexType>
<xs:simpleType name="FingerprintValueType">
  <xs:restriction base="xs:string">
    <xs:minLength value="2732"/>
    <xs:maxLength value="2732"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="FingerprintType">
  <xs:simpleContent>

```

```

    <xs:extension base="mce:FingerprintValueType">
      <xs:attribute name="id" type="xs:token" use="required"/>
      <xs:attribute name="threshold" type="mce:ProbabilityType" use="required"/>
      <xs:attribute name="shingleCount" type="xs:positiveInteger" use="required"/>
      <xs:attribute name="description" type="xs:string" use="optional"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="RegexType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="id" type="xs:token" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="KeywordType">
  <xs:sequence>
    <xs:element name="Group" type="mce:GroupType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:token" use="required"/>
</xs:complexType>
<xs:complexType name="GroupType">
  <xs:sequence>
    <xs:choice>
      <xs:element name="Term" type="mce:TermType" maxOccurs="unbounded"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="matchStyle" default="word">
    <xs:simpleType>
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="word"/>
        <xs:enumeration value="string"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="TermType">
  <xs:simpleContent>
    <xs:extension base="mce:RestrictedTermType">
      <xs:attribute name="caseSensitive" type="xs:boolean" default="false"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="LocalizedStringsType">
  <xs:sequence>
    <xs:element name="Resource" type="mce:ResourceType" maxOccurs="unbounded">
      <xs:key name="UniqueLangCodeUsedInNamePerResource">
        <xs:selector xpath="mce:Name"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
      <xs:key name="UniqueLangCodeUsedInDescriptionPerResource">
        <xs:selector xpath="mce:Description"/>
        <xs:field xpath="@langcode"/>
      </xs:key>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ResourceType">
  <xs:sequence>
    <xs:element name="Name" type="mce:ResourceNameType" maxOccurs="unbounded"/>
    <xs:element name="Description" type="mce:DescriptionType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="idRef" type="mce:GuidType" use="required"/>
</xs:complexType>
<xs:complexType name="ResourceNameType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="default" type="xs:boolean" default="false"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

```



```

        <xs:attribute name="langcode" type="mce:LangType" use="required"/>
    </xs:extension>
</xs:simpleContent>
</xs:complexType>
</xs:complexType name="DescriptionType">
    <xs:simpleContent>
        <xs:extension base="xs:string">
            <xs:attribute name="default" type="xs:boolean" default="false"/>
            <xs:attribute name="langcode" type="mce:LangType" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
</xs:schema>

```

7.9 Types Schema

This section contains the contents of the MS-OXWCONFIG-types.xsd file.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema 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/types"
  elementFormDefault="qualified" version="Exchange2016" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <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="MailTips"/>
          <xs:enumeration value="UnifiedMessagingConfiguration"/>
          <xs:enumeration value="ProtectionRules"/>
          <xs:enumeration value="PolicyNudges"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:list>
  </xs:simpleType>
  <xs:complexType name="MailTipsServiceConfiguration">
    <xs:complexContent>
      <xs:extension base="t:ServiceConfiguration">
        <xs:sequence>
          <xs:element minOccurs="1" maxOccurs="1" name="MailTipsEnabled"
            type="xs:boolean"/>
          <xs:element minOccurs="1" maxOccurs="1"
            name="MaxRecipientsPerGetMailTipsRequest" type="xs:int"/>
          <xs:element minOccurs="1" maxOccurs="1" name="MaxMessageSize"
            type="xs:int"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>

```

```

        <xs:element minOccurs="1" maxOccurs="1" name="LargeAudienceThreshold"
            type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1"
            name="ShowExternalRecipientCount" type="xs:boolean"/>
        <xs:element minOccurs="1" maxOccurs="1" name="InternalDomains"
            type="t:SmtpDomainList"/>
        <xs:element minOccurs="1" maxOccurs="1"
            name="PolicyTipsEnabled" type="xs:boolean" />
        <xs:element minOccurs="1" maxOccurs="1" name="LargeAudienceCap"
            type="xs:int" />
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="UnifiedMessageServiceConfiguration">
    <xs:complexContent>
        <xs:extension base="t:ServiceConfiguration">
            <xs:sequence>
                <xs:element minOccurs="1" maxOccurs="1" name="UmEnabled"
                    type="xs:boolean"/>
                <xs:element minOccurs="1" maxOccurs="1" name="PlayOnPhoneDialString"
                    type="xs:string"/>
                <xs:element minOccurs="1" maxOccurs="1" name="PlayOnPhoneEnabled"
                    type="xs:boolean"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfProtectionRulesType">
    <xs:sequence>
        <xs:element name="Rule" type="t:ProtectionRuleType" minOccurs="0"
            maxOccurs="unbounded"/>
    </xs:sequence>
</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:choice>
</xs:complexType>

```

```

        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:complexType name="ConfigurationRequestDetailsType">
  <xs:choice minOccurs="1" maxOccurs="1">
    <xs:any processContents="skip" minOccurs="0" maxOccurs="unbounded"
      namespace="##any"/>
  </xs:choice>
</xs:complexType>
<xs:complexType name="PolicyNudgeRulesServiceConfiguration">
  <xs:sequence>
    <xs:any processContents="skip" minOccurs="0" maxOccurs="unbounded"
      namespace="##any"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

8 Appendix C: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

- Microsoft Exchange Server 2010
- Microsoft Exchange Server 2013
- Microsoft Exchange Server 2016
- Microsoft Outlook 2010
- Microsoft Outlook 2013
- Microsoft Outlook 2016
- Microsoft Exchange Server 2019
- Microsoft Outlook 2019
- Microsoft Outlook 2021
- Microsoft Outlook LTSC 2024

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates 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 3.1.4.1.2.3](#): Exchange 2010 and Outlook 2010 do not include the **PolicyNudgeRulesConfiguration** element, as described in section [3.1.4.1.2.3](#).

<2> [Section 3.1.4.1.2.4](#): Exchange 2010 and Outlook 2010 do not include the **rule** element, as described in section [3.1.4.1.2.4](#).

<3> [Section 3.1.4.1.2.5](#): Exchange 2010 and Outlook 2010 do not include the **RulePackage** element, as described in section [3.1.4.1.2.5](#).

<4> [Section 3.1.4.1.3](#): Exchange 2010 does not use the **PolicyNudgeRulesServiceConfiguration** complex type, as described in section [3.1.4.1.3.30](#).

<5> [Section 3.1.4.1.3.1](#): Exchange 2010 and Outlook 2010 do not include the **actionsType** complex type, as described in section [3.1.4.1.3.1](#).

<6> [Section 3.1.4.1.3.2](#): Exchange 2010 and Outlook 2010 do not include the **actionTypeType** complex type, as described in section [3.1.4.1.3.2](#).

<7> [Section 3.1.4.1.3.3](#): Exchange 2010 and Outlook 2010 do not include the **AffinityType** complex type, as described in section [3.1.4.1.3.3](#).

<8> [Section 3.1.4.1.3.4](#): Exchange 2010 and Outlook 2010 do not include the **AnyType** complex type, as described in section [3.1.4.1.3.4](#).

<9> [Section 3.1.4.1.3.5](#): Exchange 2010 and Outlook 2010 do not include the **ApplyType** complex type, as described in section [3.1.4.1.3.5](#).

- <10> [Section 3.1.4.1.3.9](#): Exchange 2010 and Outlook 2010 do not include the **ClassificationDefinitionsType** complex type, as described in section [3.1.4.1.3.9](#).
- <11> [Section 3.1.4.1.3.10](#): Exchange 2010 and Outlook 2010 do not include the **ClassificationDefinitionType** complex type, as described in section [3.1.4.1.3.10](#).
- <12> [Section 3.1.4.1.3.11](#): Exchange 2010 and Outlook 2010 do not include the **classificationType** complex type, as described in section [3.1.4.1.3.11](#).
- <13> [Section 3.1.4.1.3.12](#): Exchange 2010 and Outlook 2010 do not use the **ConfigurationRequestDetailsType** complex type, as specified in section [3.1.4.1.3.12](#).
- <14> [Section 3.1.4.1.3.13](#): Exchange 2010 and Outlook 2010 do not use the **DescriptionType** complex type, as described in section [3.1.4.1.3.13](#).
- <15> [Section 3.1.4.1.3.14](#): Exchange 2010 and Outlook 2010 do not include the **DetailsType** complex type, as described in section [3.1.4.1.3.14](#).
- <16> [Section 3.1.4.1.3.15](#): Exchange 2010 and Outlook 2010 do not include the **EncryptionType** complex type, as described in section [3.1.4.1.3.15](#).
- <17> [Section 3.1.4.1.3.16](#): Exchange 2010 and Outlook 2010 do not include the **EntityType** complex type, as described in section [3.1.4.1.3.16](#).
- <18> [Section 3.1.4.1.3.17](#): Exchange 2010 and Outlook 2010 do not include the **EvidenceType** complex type, as described in section [3.1.4.1.3.17](#).
- <19> [Section 3.1.4.1.3.19](#): Exchange 2010 and Outlook 2010 do not use the **ConfigurationRequestDetails** element in the **GetServiceConfigurationType** complex type, as described in section [3.1.4.1.3.19](#).
- <20> [Section 3.1.4.1.3.20](#): Exchange 2010 and Outlook 2010 do not include the **GroupType** complex type, as described in section [3.1.4.1.3.20](#).
- <21> [Section 3.1.4.1.3.21](#): Exchange 2010 and Outlook 2010 do not include the **IdMatchType** complex type, as described in section [3.1.4.1.3.21](#).
- <22> [Section 3.1.4.1.3.22](#): Exchange 2010 and Outlook 2010 do not include the **KeywordType** complex type, as described in section [3.1.4.1.3.22](#).
- <23> [Section 3.1.4.1.3.23](#): Exchange 2010 and Outlook 2010 do not include the **localeType** complex type, as described in section [3.1.4.1.3.23](#).
- <24> [Section 3.1.4.1.3.24](#): Exchange 2010 and Outlook 2010 do not include the **LocalizedDetailsType** complex type, as described in section [3.1.4.1.3.24](#).
- <25> [Section 3.1.4.1.3.25](#): Exchange 2010 and Outlook 2010 do not include the **LocalizedStringsType** complex type, as described in section [3.1.4.1.3.25](#).
- <26> [Section 3.1.4.1.3.26](#): Exchange 2010 and Outlook 2010 do not include the **MatchType** complex type, as described in section [3.1.4.1.3.26](#).
- <27> [Section 3.1.4.1.3.27](#): Exchange 2010 and Outlook 2010 do not include the **PolicyTipsEnabled** element in the **MailTipsServiceConfiguration** complex type, as described in section [3.1.4.1.3.27](#).
- <28> [Section 3.1.4.1.3.27](#): Exchange 2010 and Outlook 2010 do not include the **LargeAudienceCap** element in the **MailTipsServiceConfiguration** complex type.
- <29> [Section 3.1.4.1.3.28](#): Exchange 2010 and Outlook 2010 do not include the **orType** complex type, as described in section [3.1.4.1.3.28](#).

<30> [Section 3.1.4.1.3.29](#): Exchange 2010 and Outlook 2010 do not include the **orType** complex type, as described in section 3.1.4.1.3.28.

<31> [Section 3.1.4.1.3.30](#): Exchange 2010 does not include the **PolicyNudgeRulesServiceConfiguration** complex type, as described in section 3.1.4.1.3.30.

<32> [Section 3.1.4.1.3.31](#): Exchange 2010 and Outlook 2010 do not include the **PolicyNudgeRulesConfigurationType** complex type, as described in section [3.1.4.1.3.31](#).

<33> [Section 3.1.4.1.3.32](#): Exchange 2010 and Outlook 2010 do not include the **PolicyNudgeRulesType** complex type, as described in section [3.1.4.1.3.32](#).

<34> [Section 3.1.4.1.3.33](#): Exchange 2010 and Outlook 2010 do not include the **PolicyNudgeRuleType** complex type, as described in section [3.1.4.1.3.33](#).

<35> [Section 3.1.4.1.3.34](#): Exchange 2010 and Outlook 2010 do not include the **predicateElementConstantType** complex type, as described in section [3.1.4.1.3.34](#).

<36> [Section 3.1.4.1.3.35](#): Exchange 2010 and Outlook 2010 do not include the **predicateElementMultipleChildType** complex type, as described in section [3.1.4.1.3.35](#).

<37> [Section 3.1.4.1.3.36](#): Exchange 2010 and Outlook 2010 do not include the **predicateElementSingleChildType** complex type, as described in section [3.1.4.1.3.36](#).

<38> [Section 3.1.4.1.3.45](#): Exchange 2010 and Outlook 2010 do not include the **PublisherType** complex type, as described in section [3.1.4.1.3.45](#).

<39> [Section 3.1.4.1.3.46](#): Exchange 2010 and Outlook 2010 do not include the **recipientType** complex type, as described in section [3.1.4.1.3.46](#).

<40> [Section 3.1.4.1.3.47](#): Exchange 2010 and Outlook 2010 do not include the **RegexType** complex type, as described in section [3.1.4.1.3.47](#).

<41> [Section 3.1.4.1.3.48](#): Exchange 2010 and Outlook 2010 do not include the **ResourceNameType** complex type, as described in section [3.1.4.1.3.48](#).

<42> [Section 3.1.4.1.3.49](#): Exchange 2010 and Outlook 2010 do not include the **ResourceType** complex type, as described in section [3.1.4.1.3.49](#).

<43> [Section 3.1.4.1.3.50](#): Exchange 2010 and Outlook 2010 do not include the **RulePackageContainerType** complex type, as described in section [3.1.4.1.3.50](#).

<44> [Section 3.1.4.1.3.51](#): Exchange 2010 and Outlook 2010 do not include the **RulePackageType** complex type, as described in section [3.1.4.1.3.51](#).

<45> [Section 3.1.4.1.3.52](#): Exchange 2010 and Outlook 2010 do not include the **RulePackType** complex type, as described in section [3.1.4.1.3.52](#).

<46> [Section 3.1.4.1.3.53](#): Exchange 2010 and Outlook 2010 do not include the **ruleType** complex type, as described in section [3.1.4.1.3.53](#).

<47> [Section 3.1.4.1.3.54](#): Exchange 2010 and Outlook 2010 do not include the **RulesType** complex type, as described in section [3.1.4.1.3.54](#).

<48> [Section 3.1.4.1.3.55](#): Exchange 2010 and Outlook 2010 do not include the **senderType** complex type, as described in section [3.1.4.1.3.55](#).

<49> [Section 3.1.4.1.3.57](#): Exchange 2010 does not use the **PolicyNudgeRulesConfiguration** element in the **ServiceConfigurationResponseMessageType** complex type, as specified in section [3.1.4.1.3.57](#).

<50> [Section 3.1.4.1.3.60](#): Exchange 2010 and Outlook 2010 do not include the **TermType** complex type.

<51> [Section 3.1.4.1.3.63](#): Exchange 2010 and Outlook 2010 do not include the **versionType** complex type.

<52> [Section 3.1.4.1.3.64](#): Exchange 2010 and Outlook 2010 do not include the **VersionedRuleType** complex type.

<53> [Section 3.1.4.1.3.65](#): Exchange 2010 and Outlook 2010 do not include the **VersionedPatternType** complex type.

<54> [Section 3.1.4.1.3.66](#): Exchange 2010 and Outlook 2010 do not include the **VersionedEvidenceType** complex type.

<55> [Section 3.1.4.1.3.67](#): Exchange 2010 and Outlook 2010 do not include the **FingerprintType** complex type.

<56> [Section 3.1.4.1.4.1](#): Exchange 2010 and Outlook 2010 do not include the **GuidType** simple type.

<57> [Section 3.1.4.1.4.2](#): Exchange 2010 and Outlook 2010 do not include the **LangType** simple type, as described in section [3.1.4.1.4.2](#).

<58> [Section 3.1.4.1.4.3](#): Exchange 2010 and Outlook 2010 do not include the **minRequiredVersion** simple type, as described in section [3.1.4.1.4.3](#).

<59> [Section 3.1.4.1.4.4](#): Exchange 2010 and Outlook 2010 do not include the **NameType** simple type, as described in section [3.1.4.1.4.4](#).

<60> [Section 3.1.4.1.4.5](#): Exchange 2010 and Outlook 2010 do not include the **OptionalNameType** simple type, as described in section [3.1.4.1.4.5](#).

<61> [Section 3.1.4.1.4.10](#): Exchange 2010 and Outlook 2010 do not include the **ProbabilityType** simple type, as described in section [3.1.4.1.4.10](#).

<62> [Section 3.1.4.1.4.11](#): Exchange 2010 and Outlook 2010 do not include the **ProximityType** simple type, as described in section [3.1.4.1.4.11](#).

<63> [Section 3.1.4.1.4.12](#): Exchange 2010 and Outlook 2010 do not include the **RestrictedTermType** simple type, as described in section [3.1.4.1.4.12](#).

<64> [Section 3.1.4.1.4.13](#): Exchange 2010 and Outlook 2010 do not include the **RulePackNameType** simple type, as described in section [3.1.4.1.4.13](#).

<65> [Section 3.1.4.1.4.14](#): Exchange 2010 does not use the **PolicyNudges** value in the **ServiceConfigurationType** simple type, as described in section [3.1.4.1.4.14](#).

<66> [Section 3.1.4.1.4.15](#): Exchange 2010 and Outlook 2010 do not include the **WorkloadType** simple type, as described in section [3.1.4.1.4.15](#).

<67> [Section 3.1.4.1.4.16](#): Exchange 2010 and Outlook 2010 do not include the **yesnoType** simple type, as described in section [3.1.4.1.4.16](#).

<68> [Section 3.1.4.1.4.17](#): Exchange 2010 and Outlook 2010 do not include the **EngineVersionType** simple type, as described in section [3.1.4.1.4.17](#).

<69> [Section 3.1.4.1.4.17](#): Exchange 2010 and Exchange 2013 do not use this engine version. It is introduced by Microsoft Exchange Server 2013 Service Pack 1 (SP1).

<70> [Section 3.1.4.1.4.18](#): Exchange 2010 and Outlook 2010 do not include the **FingerprintValueType** simple type, as described section [3.1.4.1.4.18](#).

<71> [Section 7.1](#): The classification rule package container type schema is not applicable to Exchange 2010 and Outlook 2010.

<72> [Section 7.2](#): The DLP rule schema is not applicable to Exchange 2010 and Outlook 2010.

<73> [Section 7.3](#): The DLP rule types schema is not applicable to Exchange 2010 and Outlook 2010.

<74> [Section 7.4](#): The DLP rules configuration schema is not applicable to Exchange 2010 and Outlook 2010.

<75> [Section 7.5](#): The DLP rules configuration types schema is not applicable to Exchange 2010 and Outlook 2010.

<76> [Section 7.7](#): The rule package schema is not applicable to Exchange 2010 and Outlook 2010.

<77> [Section 7.8](#): The rule package types schema is not applicable to Exchange 2010 and Outlook 2010.

9 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
- A document revision that captures changes to protocol functionality.

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 **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	Description	Revision class
8 Appendix C: Product Behavior	Updated list of supported products.	Major

10 Index

A

Abstract data model
[server](#) 14
[Applicability](#) 10
[Attribute groups](#) 13
[Attributes](#) 13

C

[Capability negotiation](#) 11
[Change tracking](#) 98
[Complex types](#) 13

D

Data model - abstract
[server](#) 14

E

Events
[local - server](#) 70
[timer - server](#) 70

Examples

[GetServiceConfiguration operation request](#) 71
[GetServiceConfiguration operation response](#) 71
[Unsuccessful GetServiceConfiguration operation error response](#) 73
[Unsuccessful GetServiceConfiguration operation response SOAP exception](#) 72

F

[Fields - vendor-extensible](#) 11
[Full WSDL](#) 75
[Full XML schema](#) 77
[Classification Rule Package Container Type Schema](#) 77
[DLP Rule Schema](#) 77
[DLP Rule Types Schema](#) 78
[DLP Rules Configuration Schema](#) 80
[DLP Rules Configuration Types Schema](#) 81
[Messages Schema](#) 82
[Rule Package Schema](#) 84
[Rule Package Types Schema](#) 84
[Types Schema](#) 89

G

[GetServiceConfiguration operation request example](#) 71
[GetServiceConfiguration operation response example](#) 71
[Glossary](#) 7
[Groups](#) 13

I

[Implementer - security considerations](#) 74
[Index of security parameters](#) 74

[Informative references](#) 9

Initialization
[server](#) 14
[Introduction](#) 7

L

Local events
[server](#) 70

M

Message processing
[server](#) 14
Messages
[attribute groups](#) 13
[attributes](#) 13
[complex types](#) 13
[elements](#) 12
[enumerated](#) 12
[groups](#) 13
[namespaces](#) 12
[simple types](#) 13
[syntax](#) 12
[transport](#) 12

N

[Namespaces](#) 12
[Normative references](#) 9

O

Operations
[GetServiceConfiguration Operation](#) 14
[Overview \(synopsis\)](#) 10

P

[Parameters - security index](#) 74
[Preconditions](#) 10
[Prerequisites](#) 10
[Product behavior](#) 93
Protocol Details
[overview](#) 14

R

[References](#) 9
[informative](#) 9
[normative](#) 9
[Relationship to other protocols](#) 10

S

Security
[implementer considerations](#) 74
[parameter index](#) 74
Sequencing rules
[server](#) 14
Server

- [abstract data model](#) 14
- [GetServiceConfiguration Operation operation](#) 14
- [initialization](#) 14
- [local events](#) 70
- [message processing](#) 14
- [sequencing rules](#) 14
- [timer events](#) 70
- [timers](#) 14
- [Simple types](#) 13
- [Standards assignments](#) 11
- Syntax
 - [messages - overview](#) 12

T

- Timer events
 - [server](#) 70
- Timers
 - [server](#) 14
- [Tracking changes](#) 98
- [Transport](#) 12
- Types
 - [complex](#) 13
 - [simple](#) 13

U

- Unsuccessful GetServiceConfiguration operation
 - response example
 - [error response](#) 73
 - [SOAP exception](#) 72

V

- [Vendor-extensible fields](#) 11
- [Versioning](#) 11

W

- [WSDL](#) 75

X

- [XML schema](#) 77
 - [Classification Rule Package Container Type Schema](#) 77
 - [DLP Rule Schema](#) 77
 - [DLP Rule Types Schema](#) 78
 - [DLP Rules Configuration Schema](#) 80
 - [DLP Rules Configuration Types Schema](#) 81
 - [Messages Schema](#) 82
 - [Rule Package Schema](#) 84
 - [Rule Package Types Schema](#) 84
 - [Types Schema](#) 89