

GT4 Delegation Service Public Interfaces

GT4 Delegation Service Public Interfaces

Table of Contents

1. APIs	1
1. Programming Model Overview	1
2. Component API	1
2. Services and WSDL	2
1. Protocol overview	2
2. Operations	2
3. Delegation Service Resource properties	2
4. Faults	2
5. WSDL and Schema Definition	2
I. Command-line tools	?
globus-credential-delegate	4
globus-credential-refresh	5
globus-delegation-client	7
wsrf-destroy	10
wsrf-query	12
3. Configuring	14
1. Configuration overview	14
2. Syntax of the interface	14
4. Environment variable interface	16
1. Environmental variables for WS Authentication & Authorization (Java)	16
A. Errors	17
Glossary	21

List of Tables

1. globus-credential-delegate options	4
2. globus-credential-refresh options	6
3. Common options	8
4. Application-specific options	8
5. Common options	11
6. Common options	13
A.1. Java WS A&A Errors	18
A.2. WS A&A Delegation Service Error Messages	20

Chapter 1. APIs

1. Programming Model Overview

This component consists of two services: the Delegation Factory Service and the Delegation Service.

The Delegation Factory Service exposes its public certificate as a resource property and allows clients to delegate credentials bound to that *public key*. Upon delegation, an Endpoint Reference(EPR) to the delegated credential, which is implemented as a resource of the Delegation Service, is returned to the client. The client can use this EPR to provide a reference to the delegated credential to other services.

The Delegation Service itself has an interface to allow refreshing the credentials remotely. Other co-hosted services can register interest in delegated credentials through listeners and be notified when credentials are refreshed.

2. Component API

Some relevant API:

- `org.globus.delegation.DelegationUtil`
- `org.globus.delegation.DelegationRefreshListener`
- `org.globus.delegation.delegationService.DelegationPortType`
- `org.globus.delegation.delegationService.DelegationFactoryPortType`

Complete API:

- [Service API](#)¹
- [Common API](#)²

¹ http://www.globus.org/api/javadoc-4.2.1/globus_wsrf_delegation_service_java/

² http://www.globus.org/api/javadoc-4.2.1/globus_wsrf_delegation_stubs_java/

Chapter 2. Services and WSDL

1. Protocol overview

The Delegation Service allows for delegation of credentials and is based on the [WS-Trust](http://www.ibm.com/developerworks/library/ws-trust/)¹ specification. A WSDL interface to refresh the credentials remotely is also provided. Access to these credentials is restricted to co-hosted services, i.e. services that are run in the same container, and is done using shared Java state. Co-hosted services interested in the credentials can register listeners and will be notified upon credential refresh.

2. Operations

2.1. Delegation Factory Service

- `RequestSecurityToken`: This operation allows for a security token to be sent to the service.

2.2. Delegation Service

- `refresh`: This operation is used to refresh a delegated credential. When invoked, all services that have registered interest in the credential through listeners are notified.

3. Delegation Service Resource properties

3.1. Delegation Factory Service

- `CertificateChain`: This resource property is used to expose the certificate used by delegation service.

4. Faults

All operations on Delegation Service and Delegation Factory Service throw `RemoteException` in case of failure.

5. WSDL and Schema Definition

- [Delegation Factory Service WSDL](http://viewcvs.globus.org/viewcvs.cgi/ws-delegation/common/schema/delegationService/delegation_factory_flattened.wsdl?rev=1.3&only_with_tag=globus_4_2_0&content-type=text/vnd.viewcvs-markup)²
- [Delegation Service WSDL](http://viewcvs.globus.org/viewcvs.cgi/ws-delegation/common/schema/delegationService/delegation_flattened.wsdl?rev=1.2&only_with_tag=globus_4_2_0&content-type=text/vnd.viewcvs-markup)³

¹ <http://www.ibm.com/developerworks/library/ws-trust/>

² http://viewcvs.globus.org/viewcvs.cgi/ws-delegation/common/schema/delegationService/delegation_factory_flattened.wsdl?rev=1.3&only_with_tag=globus_4_2_0&content-type=text/vnd.viewcvs-markup

³ http://viewcvs.globus.org/viewcvs.cgi/ws-delegation/common/schema/delegationService/delegation_flattened.wsdl?rev=1.2&only_with_tag=globus_4_2_0&content-type=text/vnd.viewcvs-markup

Command-line tools

Note the **wsrp-destroy** and **wsrp-query** commands are common Java WS Core commands.

Name

globus-credential-delegate -- Delegation client

globus-credential-delegate

Tool description

Used to contact a Delegation Factory Service and store a delegated credential. A delegated credential is created and stored in a delegated credential WS-Resource, and the Endpoint Reference(EPR) of the credential is written out to a file for further use.

Command syntax

```
globus-credential-delegate [options] <eprFilename>
```

Table 1. globus-credential-delegate options

-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-debug	Runs the client with debug message traces and error stack traces.
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-g, --delegation <mode>	Enables delegation. mode can be either ' limited ' or ' full '. Only supported with the GSI Secure Conversation authentication mechanism.
-help	Prints the usage message for the client.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-x, --proxyFilename <value>	Sets the proxy file to use as the client credential.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be ' msg ' for GSI Secure Message, or ' conv ' for GSI Secure Conversation.
-p, --protection <type>	Specifies the protection level. type can be ' sig ' for signature or ' enc ' for encryption.
-s, --service <url>	Specifies the Delegation Factory Service URL.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-y, --lifetine <value>	Lifetime of delegated credential in seconds. Default is 43200 (which is 12 hours).
-z, --authorization <type>	Specifies authorization type. type can be ' self ', ' host ', ' none ', or a string specifying the expected identity of the remote party.
<eprFilename>	Filename to write the EPR of delegated credential to.

Name

globus-credential-refresh -- Delegation refresh client

globus-credential-refresh

Tool description

Used to refresh delegated credentials pointed to by the specified EPR. A new credential is generated and the one previously created by the Delegation Service is overwritten.

Command syntax

globus-credential-refresh [options]

Table 2. globus-credential-refresh options

-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-debug	Runs the client with debug message traces and error stack traces
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference. The EPR would be of the delegation resource that is to be refreshed.
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-help	Prints the usage message for the client.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , where localPart is the simple value of the key. For complex keys, use the --eprFile option. For Delegation resource, the name will be as specified in the <i>delegationResourceKey</i> element and will replace <i>delegationResourceKey</i> with the actual key: -k " {http://www.globus.org/08/2004/delegationService}DelegationKey delegationResourceKey
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-s, --service <url>	Specifies the Delegation Factory Service URL.
-x, --proxyFileName <value>	Sets the proxy file to use as the client credential.
-y, --lifetime <value>	Lifetime of delegated credential in seconds. Defaults to 43200 (which is 12 hours).
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.

Name

globus-delegation-client -- C Delegation client

```
globus-delegation-client [OPTION...] {SERVICE-SPECIFIER} {{EPR-FILENAME} | {-refresh}}
```

Description

Create or refresh delegated credentials in a service container. If the `-refresh` option is specified on the command-line, then the credential associated with an existing `DelegationService` resource is updated with a new credential. Otherwise, the `SERVICE-SPECIFIER` is interpreted as a `DelegationFactoryService` and a new `DelegationService` resource is created.

Command syntax

```
globus-delegation-client [OPTION...] {SERVICE-SPECIFIER} {{EPR-FILENAME} | {-refresh}}
```

`SERVICE-SPECIFIER`: [-s URI [-k KEY VALUE] | -e FILENAME]

`EPR-FILENAME`: Name of file to store EPR of new delegated credential.

Table 3. Common options

-a --anonymous	Use anonymous authentication. Requires either -m 'conv' or transport (https) security.
-d, --debug	Enables debug mode. In debug mode, all SOAP messages will be displayed to stderr and full WSRF Fault messages will be displayed.
-e --eprFile FILENAME	Load service EPR from FILENAME. This EPR is used to contact the WSRF service.
-h --help	Displays help information about the command.
-k --key KEYNAME VALUE	Set resource key in the service EPR to be named KEYNAME with VALUE as its value. This can be combined with -s to construct an EPR without having an xml file on hand. The KEYNAME is a QName string in the format {namespaceURI}localPart . while the VALUE is a literal string to place in the element. For example, the option -k '{http://www.globus.org}MyKey' 128 would be rendered as <MyKey xmlns="http://www.globus.org">128</MyKey>
-m, --securityMech TYPE	Set authentication mechanism. TYPE is one of msg for WS-SecureMessage or conv for WS-SecureConversation.
-p, --protection LEVEL	Set message protection level. LEVEL is one of sig for digital signature or enc for encryption. The default is 'sig'.
-s --service ENDPOINT	Set ENDPOINT the service URL to use. Will be composed with the -k parameter if present to add ReferenceProperties to the ENDPOINT
-t --timeout SECONDS	Set client timeout to SECONDS.
-u --usage	Print short usage message.
-V --version	Show version information and exit.
-v --certKeyFiles CERTIFICATE-FILENAME KEY-FILENAME	Use credentials located in CERTIFICATE-FILENAME and KEY-FILENAME . The key file must be unencrypted.
-x --proxyFilename FILENAME	Use proxy credentials located in FILENAME .
-z --authorization TYPE	Set authorization mode. TYPE can be self , host , none , or a string specifying the identity of the remote party. The default is self .
--versions	Show version information for all loaded modules and exit.

Table 4. Application-specific options

-g --delegation MODE	Set the delegation mode. MODE can be 'limited' or 'full'. The default is 'limited'
-r --refresh	Refresh a credential instead of creating a new delegated credential resource.

Examples

Create a new delegated credential resource and store the EPR of the resource in `~/ .globus/delegation.epr`:

```
% globus-delegation-client -z host -s https://gridhost.virtual.org:8443/wsrf/services/Dele
```

Refresh the previously delegated credential:

```
% globus-delegation-client -z host -e ~/delegation.epr -refresh
```

Destroy the delegated credential:

```
% globus-wsrf-destroy -z host -e ~/delegation.epr
```

Name

`wsrf-destroy --` Destroys a resource

`wsrf-destroy`

Tool description

Destroys a resource.

Command syntax

`wsrf-destroy [options]`

Table 5. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrfl-destroy -s http://localhost:8080/wsrfl/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123
```

Name

wsrf-query -- Performs query on a resource property document

wsrf-query

Tool description

Queries the resource property document of a resource. By default, a simple XPath query is assumed that returns the entire resource property document.

Command syntax

```
wsrf-query [options] [query expression] [dialect]
```

Table 6. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: -k "{http://www.globus.org}MyKey" 123
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Examples:

```
$ wsrif-query -s https://127.0.0.1:8443/wsrif/services/DefaultIndexService \
  "count(//*[local-name()='Entry'])"
```

```
$ wsrif-query -s https://127.0.0.1:8443/wsrif/services/DefaultIndexService \
  "number(//*[local-name()='GLUECE']/glue:ComputingElement/glue:State/@glue:FreeCPUs)=0"
```

```
$ wsrif-query -s http://localhost:8080/wsrif/services/ContainerRegistryService \
  "/*/*/*/*/*[local-name()='Address']"
```

Chapter 3. Configuring

1. Configuration overview

The security settings for Delegation Factory Service and Delegation Service can be configured by modifying the [security descriptors](#). The descriptors allow for configuring the credentials that will be used by the services and the type of authentication and message protection required, as well as the authorization mechanism.

By default, the following configuration is installed:

- Delegation Factory Service:
 - Credentials are determined by the container-level security descriptor. If there is no container-level security descriptor or if it does not specify which credentials to use, then default credentials are used.
 - Authentication and message integrity protection is enforced for the `requestSecurityToken` operation. Other operations do not require authentication. This means that you may use any of GSI *Transport*, GSI Secure Message or GSI Secure Conversation when invoking the `requestSecurityToken` operation on the Delegation Factory Service.
 - Access is authorized using the gridmap mechanism and no gridmap is configured in the service by default. If a gridmap is configured in the container-level security descriptor, it is used. To configure a *grid map file* for this service, refer to instructions in the next section.
- Delegation Service
 - Credentials are determined by the container-level security descriptor. If there is no container-level security descriptor or if it does not specify which credentials to use, then default credentials are used.
 - Authentication and message integrity protection is enforced for all operations. This means that you may use any of GSI Transport, GSI Secure Message or GSI Secure Conversation when interacting with the Delegation Service.
 - Access to resources managed by the Delegation Service is managed using the gridmap mechanism. The gridmap used is resource-specific and is populated with the subject of the client that originally created the resource. This implies that only the user who delegated can access (and refresh) the delegated credential.



Note

Changing required authentication and authorization methods will require corresponding changes to the clients that contact this service.



Important

If the service is configured to use GSI Secure Transport, then container credentials are used for the handshake, irrespective of whether service-level credentials are specified.

2. Syntax of the interface

To alter the security descriptor configuration refer to [Security Descriptors](#).

To alter the security configuration of the Delegation Factory Service, edit the file `$GLOBUS_LOCATION/etc/globus_delegation_service/factory-security-config.xml`.



Note

To either specify a gridmap file different from the container level configuration or to add one if the container security descriptor does not specify one, refer to [Section 1, “Configuring Default GridMap File”](#) to add a gridmap to the Delegation Factory security descriptor.

To alter the security configuration of the Delegation Service, edit the file `$GLOBUS_LOCATION/etc/globus_delegation_service/service-security-config.xml`

Chapter 4. Environment variable interface

1. Environmental variables for WS Authentication & Authorization (Java)

Refer to [Chapter 2, Configuring](#) for environment variables. Note that the above environment variables do *not* supersede any settings provided in security descriptors.

Appendix A. Errors

Table A.1. Java WS A&A Errors




Error Code	Definition	Possible Solutions
[JWSSEC-248] Secure container requires valid credentials	This error occurs when <code>globus-start-container</code> is run without any valid credentials. Either a proxy certificate or service/host certificate needs to be configured for the container to start up.	<ol style="list-style-type: none"> 1. If you are not looking to start up a container that uses GSI Secure Transport, which is used by the container by default, use <code>globus-start-container -nosec</code>. You will be able to use insecure clients and services. However, this also implies that if you have not configured individual services with credentials, you will not be able to securely access the service. 2. If you are running a personal container, generate a proxy certificate with <code>grid-proxy-init</code>. If the proxy certificate is not in the default location, configure the container security descriptor as described in Configuring Container Security Descriptor. 3. If you want to use host certificates, configure the container security descriptor as described Configuring Credentials.
Failed to start container: Container failed to initialize [Caused by: [JWSSEC-250] Failed to load certificate/key file]	This error occurs if the file path to the container certificate and key configured are invalid.	<ol style="list-style-type: none"> 1. The path to the container certificate and key are configured in <code>\$GLOBUS_LOCATION/etc/globus_wsrf_core/global_security_descriptor.xml</code>. This file is loaded as described [here - fixme link]. Ensure that the path is correct.
Failed to start container: Container failed to initialize [Caused by: [JWSSEC-249] Failed to load proxy file]	This error occurs if container proxy file configured is invalid.	<ol style="list-style-type: none"> 1. The path to the container proxy certificates are configured in <code>\$GLOBUS_LOCATION/etc/globus_wsrf_core/global_security_descriptor.xml</code>. This file is loaded as described [here - fixme link]. Ensure that the path is correct.
Failed to start container: Container failed to initialize [Caused by: [JWSSEC-245] Error parsing file: "etc/globus_wsrf_core/global_security_descriptor.xml" [Caused by: ...]	This error occurs if the container security descriptor configured is invalid.	<ol style="list-style-type: none"> 1. The container security descriptor should conform to the Container Security Descriptor Schema.¹ 2. Refer to the "Caused by: " section for details on the specific element that is not correct.

¹ http://www.globus.org/toolkit/docs/4.2.1/security/container_security_descriptor.xsd

Error Code	Definition	Possible Solutions
[JGLOBUS-77] Unknown CA	This error occurs if the CA certificate for the credentials being used is not installed correctly.	<ol style="list-style-type: none"><li data-bbox="805 243 1325 394">1. If this issue occurs on the server side, the container is not configured with CA certificates. The container looks for trusted certificates in the default location as described Java CoG Toolkit FAQ²<li data-bbox="805 426 1325 520">2. On the server side, the trusted certificates can be configured as described in Trusted Certificates<li data-bbox="805 552 1325 646">3. On the client side, trusted certificates can be configured as described in Configuring Trusted Credentials

² <http://www.globus.org/cog/distribution/1.2/FAQ.TXT>

Table A.2. WS A&A Delegation Service Error Messages

Error Code	Definition	Possible Solutions
<p>AuthorizationException: "test DN" is not authorized to use operation: {http://www.globus.org/08/2004/delegationService}requestSecurityToken</p>	<p>This exception can occur when a client whose DN is not in the <i>grid map file</i> configured for the delegation factory service attempts to delegate (using <u>globus-credential-delegate</u>) a credential to the factory service.</p> <p> Note</p> <p>The <i>test DN</i> specified in the error message is just a placeholder and will contain the DN of the user attempting to access the credential.</p>	<p>Ensure that the client is authorized to access delegation service. This requires the client DN to be added in the gridmap file.</p>
<p>AuthorizationException: "test DN" is not authorized to use operation: {http://www.globus.org/08/2004/delegationService}refresh</p>	<p>This exception can occur when a client attempts to refresh a credential it did not delegate (using <u>globus-credential-refresh</u>).</p> <p> Note</p> <p>The <i>test DN</i> specified in the error message is just a placeholder and will contain the DN of the user attempting to access the credential.</p>	<p>This is a delegation service policy and only client who delegates can refresh the credential.</p>
<p><i>test user DN</i> is not authorized to access this resource</p>	<p>Similar to above error but experienced by developers using the API - Only the user who created the delegated credential is allowed to access it. There are two sets of API functions for getting the credential and registering listeners: one in which the caller's DN is picked up from the current thread and the other in which a JAAS subject (containing the caller's DN) is explicitly passed as a function parameter. If the caller's DN (picked up from thread or specified explicitly) does not match the DN of the user who created the credential, this error is thrown.</p> <p> Note</p> <p>The <i>test DN</i> specified in the error message is just a placeholder and will contain the DN of the user attempting to access the credential.</p>	<p>Ensure that the DN explicitly specified or the client DN associated with the thread matches the creator's DN.</p>
<p>Unable to retrieve caller DN, cannot register</p>	<p>Developers come across this error when attempting to register a listener with a delegated credential resource without a JAAS subject. There are two ways of registering: either the JAAS subject can be explicitly passed using the API or the JAAS subject can be picked up from the current message context (the subject representing the client). If the latter mechanism for registering is used and there is no client credential associated with the thread that is calling the register function, then this exception is thrown.</p>	<p>Make sure to use the API call that explicitly passes the subject.</p>

Glossary

C

certificate A public key plus information about the certificate owner bound together by the digital signature of a CA. In the case of a CA certificate, the certificate is self signed, i.e. it was signed using its own private key.

G

grid map file A file containing entries mapping certificate subjects to local user names. This file can also serve as a access control list for GSI enabled services and is typically found in `/etc/grid-security/grid-mapfile`. For more information see the Gridmap section [here](#).

P

public key The public part of a key pair used for cryptographic operations (e.g. signing, encrypting).

T

transport-level security Uses transport-level security (TLS) mechanisms.

W

Web Services Addressing (WSA) The WS-Addressing specification defines transport-neutral mechanisms to address web services and messages. Specifically, it defines XML elements to identify web service endpoints and to secure end-to-end endpoint identification in messages. See the [W3C WS Addressing Working Group](#)¹⁴ for details.

X

XML Extensible Markup Language (XML) is standard, flexible, and extensible data format used for web services. See the [W3C XML site](#)²⁰ for details.

¹⁴ <http://www.w3.org/2002/ws/addr/>

²⁰ <http://www.w3.org/XML/>