

GT4 C WS A&A User's Guide

DRAFT

GT4 C WS A&A User's Guide

Introduction

Typical user configuration for this component deals with configuring authentication mechanisms and credentials for the clients. These could be client applications, including command line clients or client configuration within services that contact other services.

There are multiple mechanisms for doing this:

- Command line options (these are application-specific)
- Client security descriptors
- CoG properties
- Environment variables
- Relying on default behavior. The only default behaviors available concern the proxy file and trusted certificates locations.

More information on these mechanisms can be found in the [public interface guide](#).

DRAFT

Table of Contents

| | |
|--------------------------------------|----|
| I. Command-line tools | 5 |
| globus-credential-delegate | 6 |
| 1. Domain-specific interface | 1 |
| 1. Interface introduction | 1 |
| 2. Syntax of the interface | 2 |
| 2. Debugging | 7 |
| 1. Logging | 7 |
| 3. Troubleshooting | 8 |
| 1. Credential Troubleshooting | 8 |
| 2. Error Messages For C WS A&A | 11 |
| Glossary | 16 |

DRAFT

List of Tables

| | |
|---|----|
| 1. globus-credential-delegate options | 6 |
| 1.1. Client side security properties | 3 |
| 3.1. Credential Errors | 9 |
| 3.2. C WS A&A Errors | 12 |

DRAFT

Command-line tools

DRAFT

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

| | |
|-----------|--|
| [option1] | Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism. |
| [option1] | Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism. |

Chapter 1. Domain-specific interface

1. Interface introduction

Client-side security is set up by setting individual properties on the `javax.xml.rpc.Stub` object used for the web service method invocation or by setting properties on a client-side security descriptor object, which in turn is propagated to client-side security handlers by making it available as a stub object property. Here are examples of the two approaches:

- Setting a property on the stub:

```
// Create endpoint reference
EndpointReferenceType endpoint = new EndpointReferenceType();
// Set address of service
String counterAddr =
    "http://localhost:8080/wsrf/services/CounterService";
// Get handle to port
CounterPortType port =
    locator.getCounterPortTypePort(endpoint);
// set client authorization to self
((Stub)port)._setProperty(Constants.AUTHORIZATION,
    SelfAuthorization.getInstance());
```

- Setting properties using a client descriptor:

```
// Client security descriptor file
String CLIENT_DESC =
    "org/globus/wsrf/samples/counter/client/client-security-config.xml";
// Create endpoint reference
EndpointReferenceType endpoint = new EndpointReferenceType();
// Set address of service
String counterAddr =
    "http://localhost:8080/wsrf/services/CounterService";
// Get handle to port
CounterPortType port =
    locator.getCounterPortTypePort(endpoint);
//Set descriptor on Stub
((Stub)port)._setProperty(Constants.CLIENT_DESCRIPTOR_FILE, CLIENT_DESC);
```

Note

If the client needs to use transport security, the following API must be used to register the Axis transport handler for https:

```
import org.globus.axis.util.Util;
static {
    Util.registerTransport();
}
```

2. Syntax of the interface

DRAFT

Table 1.1. Client side security properties

| Number | Task | Stub Configuration |
|--------|---|---|
| 1. | Allows for configuration of credentials for authentication. | Property: <code>org.globus.axis.gsi.GSIConstants.GSI_CREDENTIALS</code> Value equals the Instance of <code>org.ietf.jgss.GSSCredential</code> . |
| 2. | Allows for configuring client-side authorization. | Property: <code>org.globus.wsrf.security.Constants.AUTHORIZATION</code> Value equals the Instance of <code>org.globus.wsrf.security.authorization.Authorization</code> If GSI Secure Transport or GSI Secure Conversation is used, the value should be an instance of <code>org.globus.gsi.gssapi.auth.Authorization</code> . But this translation is done automatically by the toolkit. |
| 3. | Enable GSI Secure Conversation with specified message protection level. | 1. Property: <code>org.globus.wsrf.security.Constants.GSI_SEC_CONV</code> Values equal one of the following: <ul style="list-style-type: none"> • <code>Constants.ENCRYPTION</code> • <code>Constants.SIGNATURE</code> 2. Property: <code>org.globus.wsrf.security.Constants.GSI_SEC_CONV_SECREPLY_UNNECESSARY</code> If the value is set to <code>Boolean.TRUE</code> , the GSI Secure conversation protection is not required in the reply message. By default, if the request was secured with GSI Secure Conversation, the response is also required to have the same protection. 3. Property: You can set the SOAP Actor of the GSI signed/encrypted SOAP message by using the <code>gssActor</code> property. We recommend that you <i>not</i> do this unless you <i>really</i> know what you are doing. |

| | | |
|----|---|---|
| 4. | Sets the GSI delegation mode. <i>Used for GSI Secure Conversation only.</i> If limited or full delegation is chosen, then some form of client-side authorization needs to be done (i.e client-side authorization cannot be set to none). | <p>Property:</p> <pre>org.globus.axis.gsi.GSIConstants.GSI_MODE</pre> <p>Value equals one of following:</p> <ol style="list-style-type: none"> 1. <code>GSIConstants.GSI_MODE_NO_DELEG</code>: No delegation is performed. 2. <code>GSIConstants.GSI_MODE_LIMITED_DELEG</code>: Limited delegation is performed. 3. <code>GSIConstants.GSI_MODE_FULL_DELEG</code>: Full delegation is performed. |
| 5. | Enables GSI Secure Transport with some protection level. | <p>Property:</p> <pre>org.globus.gsi.GSIConstants.GSI_TRANSPORT</pre> <p>Values equal one of the following:</p> <ul style="list-style-type: none"> • <code>Constants.ENCRYPTION</code> • <code>Constants.SIGNATURE</code> |
| 6. | Enables anonymous authentication. <i>This option only applies to GSI Secure Conversation and GSI Transport.</i> | <p>Property:</p> <pre>org.globus.wsrp.security.Constants.GSI_ANONYMOUS</pre> <p>Value equals one of following:</p> <ol style="list-style-type: none"> 1. <code>Boolean.FALSE</code>: Anonymous authentication is disabled. 2. <code>Boolean.TRUE</code>: Anonymous authentication is enabled. |

| | | |
|----|--|---|
| 7. | Enable GSI Secure Message with specified message protection level. | <p>1. Property: <code>org.globus.wsrp.security.Constants.GSI_SEC_MSG</code></p> <p>Values equal one of the following:</p> <ul style="list-style-type: none"> • <code>Constants.ENCRIPTION</code> • <code>Constants.SIGNATURE</code> <p>2. Property: <code>org.globus.wsrp.security.Constants.GSI_SEC_MSG_SECREPLY_UNNECESSARY</code></p> <p>If the value is set to <code>Boolean.TRUE</code>, the GSI Secure Message protection is not required in the reply message. By default, if the request was secured with GSI Secure Message, the response is also required to have the same protection.</p> <p>3. Property: <code>org.globus.wsrp.security.Constants.GSI_SEC_MSG_SINGLECERT</code></p> <p>If the value is set to <code>Boolean.TRUE</code>, only a single certificate is used for the GSI Secure Message request. By default, the whole certificate chain is sent.</p> <p>4. Property:</p> <p>You can set the SOAP Actor of the signed message using the <code>x509Actor</code> property, but we do <i>not</i> recommend this unless you know what you are doing.</p> |
| 8. | Enable WS-Security username/password authentication. | <p>Properties:</p> <p><code>org.globus.wsrp.security.Constants.USERNAME</code></p> <p>Value equals the username.</p> <p><code>org.globus.wsrp.security.Constants.PASSWORD</code></p> <p>Value equals the password.</p> |

| | | |
|-----|--|--|
| 9. | Sets the credential that is used to encrypt the message (typically, the recipient's <i>public key</i>). <i>Used for GSI Secure Message only.</i> | <p>Property:</p> <pre>org.globus.wsrfl.impl.security.authentication .Constants.PEER_SUBJECT</pre> <p>Value equals the instance of <code>javax.security.auth.Subject</code>.</p> <p>The credential object needs to be wrapped in <code>org.globus.wsrfl.impl.security.authentication.encryption</code> and added to the set of public credentials of the Subject object.</p> <p>For example, if <code>publicKeyFilename</code> was the file that had the recipient's public key:</p> <pre>Subject subject = new Subject(); X509Certificate serverCert = CertUtil.loadCertificate(publicKeyFilename); EncryptionCredentials encryptionCreds = new EncryptionCredentials(new X509Certificate[] { serverCert }); subject.getPublicCredentials().add(encryptionCreds); stub._setProperty(Constants.PEER_SUBJECT, subject);</pre> |
| 10. | Sets the trusted certificates location. | <p>Property:</p> <pre>org.globus.wsrfl.security.TRUSTED_CERTIFICATES</pre> <p>Value should be a comma-separated list of directories and file names.</p> |
| 11. | Sets the SAML Authorization Assertion to embed in SOAP Header. | <p>Property:</p> <pre>org.globus.wsrfl.impl.security.authentication.Constants.SAML_AUTHZ_ASSERTION</pre> <p>Value should be an instance of <code>org.opensaml.SAMLAssertion</code>.</p> |

Can
con
usin
des

Chapter 2. Debugging

1. Logging

As of 4.2.0, the Globus Toolkit provides system administration logs that are CEDPs best practices¹ compliant.

To enable CEDPS logging, pass the `-log PATH` parameter to the **globus-wsc-container** program.

For more details on the CEDPS Logging format, including descriptions of reserved name-value pairs, see <http://cedps.net/index.php/LoggingBestPractices>:

1.1. Sample log file

The sample log file² contains many log entries for various scenarios in the C WS container.

¹ <http://cedps.net/index.php/LoggingBestPractices>

² <http://www.globus.org/toolkit/docs/4.2/4.2.0/common/cwscore/sample-container-log.txt>

Chapter 3. Troubleshooting

For a list of common errors in GT, see [Error Codes](#). For information about system administrator logs, see [Chapter 7, Troubleshooting](#) in the C WS Security Admin Guide.

1. Credential Troubleshooting

1.1. Credential Errors

The following are some common problems that may cause clients or servers to report that credentials are invalid:

For a list of common errors in GT, see [Error Codes](#).

DRAFT

Table 3.1. Credential Errors

| Error Code | Definition | Possible Solutions |
|--|--|---|
| Your proxy credential may have expired | Your proxy credential may have expired. | Use <code>grid-proxy-info</code> to check whether the proxy credential has actually expired. If it has, generate a new proxy with <code>grid-proxy-init</code> . |
| The system clock on either the local or remote system is wrong. | This may cause the server or client to conclude that a credential has expired. | Check the system clocks on the local and remote system. |
| Your end-user certificate may have expired | Your end-user certificate may have expired | Use <code>grid-cert-info</code> to check your certificate's expiration date. If it has expired, follow your CA's procedures to get a new one. |
| The permissions may be wrong on your proxy file | If the permissions on your proxy file are too lax (for example, if others can read your proxy file), Globus Toolkit clients will not use that file to authenticate. | You can "fix" this problem by changing the permissions on the file or by destroying it (with <code>grid-proxy-destroy</code>) and creating a new one (with <code>grid-proxy-init</code>). Important: However, it is still possible that someone else has made a copy of that file during the time that the permissions were wrong. In that case, they will be able to impersonate you until the proxy file expires or your permissions or end-user certificate are revoked, whichever happens first. |
| The permissions may be wrong on your private key file | If the permissions on your end user certificate private key file are too lax (for example, if others can read the file), <code>grid-proxy-init</code> will refuse to create a proxy certificate. | You can "fix" this by changing the permissions on the private key file. Important: However, you will still have a much more serious problem: it is possible that someone has made a copy of your private key file. Although this file is encrypted, it is possible that someone will be able to decrypt the private key, at which point they will be able to impersonate you as long as your end user certificate is valid. You should contact your CA to have your end-user certificate revoked and get a new one. |
| The remote system may not trust your CA | The remote system may not trust your CA | Verify that the remote system is configured to trust the CA that issued your end-entity certificate. See Installing GT 4.2.0 for details. |
| You may not trust the remote system's CA | You may not trust the remote system's CA | Verify that your system is configured to trust the remote CA (or that your environment is set up to trust the remote CA). See Installing GT 4.2.0 for details. |
| There may be something wrong with the remote service's credentials | There may be something wrong with the remote service's credentials | It is sometimes difficult to distinguish between errors reported by the remote service regarding your credentials and errors reported by the client interface regarding the remote service's credentials. If you cannot find anything wrong with your credentials, check for the same conditions on the remote system (or ask a remote administrator to do so). |

1.2. Some tools to validate certificate setup

1.2.1. Check that the user certificate is valid

```
openssl verify -CApath /etc/grid-security/certificates
-purpose sslclient ~/.globus/usercert.pem
```

1.2.2. Connect to the server using s_client

```
openssl s_client -ssl3 -cert ~/.globus/usercert.pem -key
~/.globus/userkey.pem -CApath /etc/grid-security/certificates
-connect <host:port>
```

Here `<host:port>` denotes the server and port you connect to.

If it prints an error and puts you back at the command prompt, then it typically means that the *server* has closed the connection, i.e. that the server was not happy with the client's certificate and verification. Check the SSL log on the server.

If the command "hangs" then it has actually opened a telnet style (but secure) socket, and you can "talk" to the server.

You should be able to scroll up and see the subject names of the server's verification chain:

```
depth=2 /DC=net/DC=ES/O=ESnet/OU=Certificate Authorities/CN=ESnet Root CA 1
verify return:1
depth=1 /DC=org/DC=DOEGrids/OU=Certificate Authorities/CN=DOEGrids CA 1
verify return:1
depth=0 /DC=org/DC=doegrids/OU=Services/CN=wiggum.mcs.anl.gov
verify return:1
```

In this case, there were no errors. Errors would give you an extra line next to the subject name of the certificate that caused the error.

1.2.3. Check that the server certificate is valid

Requires root login on server:

```
openssl verify -CApath /etc/grid-security/certificates -purpose sslserver
/etc/grid-security/hostcert.pem
```

2. Error Messages For C WS A&A

DRAFT

Table 3.2. C WS A&A Errors

| Error Code | Definition |
|--|-------------------------------|
| <pre>ERROR: Couldn't read user key: Bad passphrase key file location: /Users/bester/.globus/userkey.pem globus_credential: Error reading user credential: Can't read credential's private key from PEM OpenSSL Error: pem_lib.c:423: in library: PEM routines, function PEM_do_header: bad decrypt OpenSSL Error: evp_enc.c:509: in library: digital envelope routines, function EVP_DecryptFinal: bad decrypt Use -debug for further information.</pre> | Unable to decrypt private key |
| <pre>globus_gsi_gssapi: Error with gss credential handle globus_credential: Valid credentials could not be found in any of the possible locations specified by the credential search order. Valid credentials could not be found in any of the possible locations specified by the credential search order. Attempt 1 globus_credential: Error reading host credential globus_sysconfig: Error with certificate filename globus_sysconfig: Error with certificate filename globus_sysconfig: File is not owned by current user: /etc/grid-security/hostcert.pem is not owned by current user Attempt 2 globus_credential: Error reading proxy credential globus_sysconfig: Could not find a valid proxy certificate file location globus_sysconfig: Error with key filename globus_sysconfig: File does not exist: /tmp/x509up_u501 is not a valid file Attempt 3 globus_credential: Error reading user credential globus_credential: Key is password protected: GSI does not currently support password protected private keys. OpenSSL Error: pem_lib.c:401: in library: PEM routines, function PEM_do_header: bad password read</pre> | No user proxy could be found |
| <pre>globus_gsi_gssapi: Error with GSI credential globus_gsi_gssapi: Error with gss credential handle globus_credential: Error with credential: The proxy credential: /tmp/x509up_u1499 with subject: /DC=org/DC=example/DC=grid/OU=People/CN=Joe User/CN=1235439010 expired 44 minutes ago.</pre> | Proxy has expired. |

| Error Code | Definition |
|---|---|
| <p>globus_xio: The GSI XIO driver failed to establish a secure connection. The failure occurred during a handshake read. globus_xio: An end of file occurred</p> | <p>Communication disrupted during SSL handshake</p> |
| <p>globus_gsi_gssapi: Unable to verify remote side's credentials globus_gsi_gssapi: Unable to verify remote side's credentials: Couldn't verify the remote certificate OpenSSL Error: s3_pkt.c:1052: in library: SSL routines, function SSL3_READ_BYTES: sslv3 alert bad certificate SSL alert number 42</p> | <p>Unable to verify remote certificate. Often a clock-synchronization problem where the service clock is behind that of the client.</p> |
| <p>OpenSSL Error: s3_clnt.c:894: in library: SSL routines, function SSL3_GET_SERVER_CERTIFICATE: certificate verify failed globus_gsi_callback_module: Could not verify credential globus_gsi_callback_module: The certificate is not yet valid: Cert with subject: /DC=org/DC=example/DC=grid/OU=People/CN=Joe User/CN=464555355 is not yet valid- check clock skew between hosts.</p> | <p>Unable to verify remote certificate. Often a clock-synchronization problem where the client clock is behind that of the service.</p> |

| Error Code | Definition |
|---|--|
| <pre> globus_gsi_callback_module: Error with signing policy globus_sysconfig: Error getting signing policy file globus_sysconfig: File does not exist: /etc/grid-security/certificates/2b0e42b2.signing_policy is not a valid file </pre> | <p>The service's certificate is not trusted by the client</p> |
| <pre> globus_gsi_callback_module: Could not verify credential globus_gsi_callback_module: Error with signing policy globus_gsi_callback_module: Error in OLD GAA code: CA policy violation: <no reason given> </pre> | <p>Service certificate is not trusted because the CA signing policy does not trust the CA to sign the subject name of the certificate.</p> |
| <pre> Error: globus_soap_message_module: SOAP Fault Fault code: Client Fault string: globus_handler_ws_secure_message: Server Request handling failed globus_handler_ws_secure_message: Failed to verify the message: Unable to get Security header element from message attributes. </pre> | <p>The client sent a request to a service which message security without properly invoking the security handlers</p> |

| Error Code | Definition |
|--|--|
| <p>Error: globus_soap_message_module: SOAP Fault Fault code: Client Fault string: globus_soap_message_module: Loaded message handlers do not understand required header element: {http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}Security</p> | <p>The client sent a request protected with message-level security but the server did not understand the required security headers</p> |

Glossary

some terms not in the docs but wanted in glossary: scheduler

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.

P

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

S

scheduler Term used to describe a job scheduler mechanism to which GRAM interfaces. It is a networked system for submitting, controlling, and monitoring the workload of batch jobs in one or more computers. The jobs or tasks are scheduled for execution at a time chosen by the subsystem according to an available policy and availability of resources. Popular job schedulers include Portable Batch System (PBS), Platform LSF, and IBM LoadLeveler.