

GT 4.2.0 MyProxy: System Administrator's Guide

DRAFT

GT 4.2.0 MyProxy: System Administrator's Guide

Introduction

This guide contains advanced configuration information for system administrators working with MyProxy. It provides references to information on procedures typically performed by system administrators, including installation, configuring, deploying, and testing the installation.

Important

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [Installing GT 4.2.0](#). Read through this guide before continuing!

A typical MyProxy configuration has one dedicated myproxy-server for the site, with MyProxy clients installed on all systems where other Globus Toolkit client software is installed.

Table of Contents

- 1. Building and Installing 1
 - 1. Building and Installing only MyProxy 1
- 2. Configuring 2
- 3. Deploying 5
- 4. Testing 6
- 5. Security Considerations 7
 - 1. MyProxy Security Considerations 7
- 6. Debugging 8
 - 1. Logging 8
- 7. Troubleshooting 9
 - 1. Incorrect system clocks 9
 - 2. Errors 10
- Glossary 11

DRAFT

List of Tables

2.1. myproxy-server.config lines	3
7.1. MyProxy Errors	10

DRAFT

Chapter 1. Building and Installing

MyProxy is built and installed as part of a default GT 4.2.0 installation. For basic installation instructions, see the [Installing GT 4.2.0](#). No extra installation steps are required for this component.

1. Building and Installing only MyProxy

If you wish to install MyProxy without installing the rest of the Globus Toolkit, follow the instructions in the [Installing GT 4.2.0](#) with the following changes. First, you do not need Ant, a JDK, or a JDBC database to build only MyProxy. Second, instead of running "make", run:

```
globus$ make gsi-myproxy
```

This will install the MyProxy client and server programs. For client-only installations, simply do not configure or use the installed server.

Chapter 2. Configuring

No additional configuration is required to use MyProxy clients after they are installed, although you may want to set the MYPROXY_SERVER environment variable to the hostname of your myproxy-server in the default user environment on your systems.

To configure the myproxy-server you must modify `$GLOBUS_LOCATION/etc/myproxy-server.config`. *If you skip this step, your myproxy-server will not accept any requests.* The default configuration does not enable any myproxy-server features to provide the greatest security until you have configured your server. To enable all myproxy-server features uncomment the provided sample policy at the top of the `myproxy-server.config` config file, as follows:

```
#
# Complete Sample Policy
#
# The following lines define a sample policy that enables all
# myproxy-server features. See below for more examples.
accepted_credentials "*"
authorized_retrievers "*"
default_retrievers "*"
authorized_renewers "*"
default_renewers "none"
```

Please see below for additional documentation on the `myproxy-server.config` options.

If you have root access, you can copy your `myproxy-server.config` file to `/etc/myproxy-server.config` so it is not overwritten by later installations.

The `myproxy-server.config` file sets the policy for the **myproxy-server(8)**, specifying what credentials may be stored in the server's repository and who is authorized to retrieve credentials. By default, the **myproxy-server(8)** looks for this file in `/etc/myproxy-server.config` and if it is not found there, it looks in `$GLOBUS_LOCATION/etc/myproxy-server.config`. The **myproxy-server -c** option can be used to specify an alternative location. The file installed by default does not allow any requests.

The file also supports a **passphrase_policy_program** command for specifying an external program for evaluating the quality of users' passphrases. A sample program is installed in `$GLOBUS_LOCATION/share/myproxy/myproxy-passphrase-policy` but is not enabled by default.

Lines in the configuration file use limited regular expressions for matching the distinguished names (DNs) of classes of users. The limited regular expressions support the shell-style characters '*' and '?', where '*' matches any number of characters and '?' matches any single character.

The DN limited regexes should be delimited with double quotes ("DN regex").

The configuration file has the following types of lines:

Table 2.1. myproxy-server.config lines

accepted_credentials "DNregex"	Each of these lines allows any clients whose DNs match the given limited regex to connect to the myproxy-server and store credentials with it for future retrieval. Any number of these lines may appear. For backwards compatibility, these lines can also start with <code>allowed_clients</code> instead of <code>accepted_credentials</code> .
authorized_retrievers "DN regex"	Each of these lines allows the server administrator to set server-wide policies for authorized retrievers. If the client DN does not match the given limited regex, the client is not allowed to retrieve the credentials previously stored by a client. In addition to the server-wide policy, MyProxy also provides support for per-credential policy. The user can specify the regex DN of the allowed retrievers of the credential when uploading the credential (using myproxy-init(1)). The retrieval client DN must also match the user specified regex. In order to retrieve credentials the client also needs to know the name and pass phrase provided by the client when the credentials were stored. Any number of these lines may appear. For backwards compatibility, these lines can also start with <code>allowed_services</code> instead of <code>authorized_retrievers</code> .
default_retrievers "DN regex"	Each of these lines allows the server administrator to set server-wide default policies. The regex specifies the clients who can access the credentials. The default retriever policy is enforced if a per-credential policy is not specified on upload (using myproxy-init(1)). In other words, the client can override this policy for a credential on upload. The per-credential policy is enforced in addition to the server-wide policy specified by the <code>authorized_retrievers</code> line (which clients can not override). Any number of these lines may be present. For backwards compatibility, if no <code>default_retrievers</code> line is specified, the default policy is "*", which allows any client to pass the per-credential policy check. (The client must still pass the <code>authorized_retrievers</code> check).
authorized_renewers "DN regex"	Each of these lines allows the server administrator to set server-wide policies for authorized renewers. If the client DN does not match the given limited regex the client is not allowed to renew the credentials previously stored by a client. In addition to the server-wide policy, MyProxy also provides support for per-credential policy. The user can specify the regex DN of the allowed renewers of the credential on upload (using myproxy-init(1)). The renewal client DN must match both this regex and the user specified regex. In this case, the client must also already have a credential with a DN matching the DN of the credentials to be retrieved, to be used in a second authorization step (see the <code>-a</code> option for myproxy-logon(1)).
default_renewers "DN regex"	Each of these lines allows the server administrator to set server-wide default renewer policies. The regex specifies the clients who can renew the credentials. The default renewer policy is enforced if a per-credential policy is not specified on upload (using myproxy-init(1)). This is enforced in addition to the server-wide policy specified by the <code>authorized_renewers</code> line. Any number of these lines may appear. For backwards compatibility, if no <code>default_renewers</code> line is specified, the default policy is "*", which allows any client to pass the per-credential policy check. (The client must still pass the <code>authorized_renewers</code> check).
passphrase_policy_program full-path-to-script	This line specifies a program to run whenever a passphrase is set or changed for implementing a local password policy. The program is passed the new passphrase via stdin and is passed the following arguments: username, distinguished name, credential name (if any), per-credential retriever policy (if any), and per-credential renewal policy (if any). If the passphrase is acceptable, the program should exit with status 0. Otherwise, it should exit with non-zero status, causing the operation in progress (credential load, passphrase change) to fail with the error message provided by the program's stdout. Note: You must specify the full path to the external program. <code>\$GLOBUS_LOCATION</code> can't be used in the <code>myproxy-server.config</code> file.

max_proxy_lifetime hours	This line specifies a server-wide maximum lifetime for retrieved proxy credentials. By default, no server-wide maximum is enforced. However, if this option is specified, the server will limit the lifetime of any retrieved proxy credentials to the value given.
-----------------------------	---

DRAFT

Chapter 3. Deploying

A sample SysV-style boot script for MyProxy is installed at `$GLOBUS_LOCATION/share/myproxy/etc.init.d.myproxy`. To install on Linux, copy the file to `/etc/rc.d/init.d/myproxy` and run:

```
chkconfig --add myproxy
```

You will need to edit the file to set the `GLOBUS_LOCATION` environment variable correctly.

Alternatively, to run the myproxy server out of `inetd` or `xinetd`, you need to do the following as root:

- Add the entries in `$GLOBUS_LOCATION/share/myproxy/etc.services.modifications` to the `/etc/services` or `/etc/inet/services` file.
- Add the entries in `$GLOBUS_LOCATION/share/myproxy/etc.inetd.conf.modifications` to `/etc/inetd.conf` or `/etc/inet/inetd.conf`, or copy `$GLOBUS_LOCATION/share/myproxy/etc.xinetd.myproxy` to `/etc/xinetd.d/myproxy`. You'll need to modify the paths in the file according to your installation.
- Reactivate the `inetd` (or `xinetd`). This is typically accomplished by sending the `SIGHUP` signal to the daemon. Refer to the `inetd` or `xinetd` man page for your system.

Chapter 4. Testing

To verify your myproxy-server installation and configuration, you can run the myproxy-server directly from your shell. If using a *host certificate*, you will need to run the myproxy-server as root. First, make sure your Globus environment is setup in your shell. Set the GLOBUS_LOCATION environment variable to the location of your MyProxy installation. Then, depending on your shell, run one of the following commands.

For csh shells:

```
source $GLOBUS_LOCATION/etc/globus-user-env.csh
```

For sh shells:

```
.$GLOBUS_LOCATION/etc/globus-user-env.sh
```

Then, run `$GLOBUS_LOCATION/sbin/myproxy-server -d`. The `-d` argument runs the myproxy-server in debug mode. It will write debugging messages to the terminal and exit after servicing a single request. You will need to start it once for each test request. In another shell, you can run the MyProxy client programs to test the server.

If run without the `-d` argument, the myproxy-server program will start up and background itself. It accepts connections on TCP port 7512, forking off a separate child to handle each incoming connection. It logs information via the syslog service under the daemon facility.

Chapter 5. Security Considerations

1. MyProxy Security Considerations

You should choose a well-protected host to run the myproxy-server on. Consult with security-aware personnel at your site. You want a host that is secured to the level of a Kerberos KDC, that has limited user access, runs limited services, and is well monitored and maintained in terms of security patches.

For a typical myproxy-server installation, the host on which the myproxy-server is running must have `/etc/grid-security` created and a *host certificate* installed. In this case, the myproxy-server will run as root so it can access the host certificate and key.

DRAFT

Chapter 6. Debugging

1. Logging

When troubleshooting a MyProxy problem, it is important to consult the myproxy-server logs. If you don't have access to the myproxy-server logs, please contact your myproxy-server administrator for help. The myproxy-server logs to the system logger (syslog) LOG_DAEMON facility. Alternatively, run

```
myproxy-server -d
```

from a terminal. In that mode, the myproxy-server will write debugging messages to the terminal and exit after servicing a single request.

As of 4.2.0, the Globus Toolkit provides system administration logs that are [CEDPs best practices](#)¹ compliant.

Configuration for this logger can be changed by editing `$GLOBUS_LOCATION/FIXME/path/to/cedpslogfile`.

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

1.1. Configuring system administration logs

[FIXME the following is java core's info - tailor to this component] The specific logger to edit will be `log4j.logger.sysadmin` in `container-log4j.properties`. There you can configure the following properties:

```
log4j.appender.infoCategory=org.apache.log4j.RollingFileAppender
log4j.appender.infoCategory.Threshold=INFO
log4j.appender.infoCategory.File=var/containerLog
log4j.appender.infoCategory.MaxFileSize=10MB
log4j.appender.infoCategory.MaxBackupIndex=2
```

Above implies the logging file is rolling with each file size limited to 10MB and the logging information is stored in `$GLOBUS_LOCATION/var/containerLog`.

1.2. Sample log file

The [sample log file](#)² contains many log entries for various scenarios in the Java WS container [FIXME does this apply for your component? if not, can you provide a sample log file?].

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

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

Chapter 7. Troubleshooting

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

1. Incorrect system clocks

The most common cause of MyProxy authentication problems is incorrect system clocks. GSI authentication is very sensitive to clock skew. Make sure your system clock is accurate (for example, by running [NTP](#)¹) and your timezone is set correctly.

¹ <http://www.ntp.org/>

2. Errors

Table 7.1. MyProxy Errors

Error Code	Definition	Possible Solutions
MyProxy server name does not match expected name	<p>This error appears as a mutual authentication failure or a server authentication failure, and the error message should list two names: the expected name of the MyProxy server and the actual authenticated name.</p> <p>By default, the MyProxy clients expect the MyProxy server to be running with a host certificate that matches the target hostname. This error can occur when running the MyProxy server under a non-host certificate or if the server is running on a machine with multiple hostnames.</p> <p>The MyProxy clients authenticate the identity of the MyProxy server to avoid sending pass-phrases and credentials to rogue servers.</p> <p>If the expected name contains an IP address, your system is unable to do a reverse lookup on that address to get the canonical hostname of the server, indicating either a problem with that machine's DNS record or a problem with the resolver on your system.</p>	<p>If the server name shown in the error message is acceptable, set the <code>MYPROXY_SERVER_DN</code> environment variable to that name to resolve the problem.</p>
Error in <code>bind()</code> : Address already in use	<p>This error indicates that the <code>myproxy-server</code> port (default: 7512) is in use by another process, probably another <code>myproxy-server</code> instance. You cannot run multiple instances of the <code>myproxy-server</code> on the same network port.</p>	<p>If you want to run multiple instances of the <code>myproxy-server</code> on a machine, you can specify different ports with the <code>-p</code> option, and then give the same <code>-p</code> option to the MyProxy commands to tell them to use the <code>myproxy-server</code> on that port.</p>
<code>grid-proxy-init</code> failed	<p>This error indicates that the <code>grid-proxy-init</code> command failed when <code>myproxy-init</code> attempted to run it, which implies a problem with the underlying Globus installation.</p>	<p>Run</p> <pre>grid-proxy-init -debug -verify</pre> <p>for more information.</p>
User not authorized	<p>An error from the <code>myproxy-server</code> saying you are "not authorized" to complete an operation typically indicates that the <code>myproxy-server.config</code> file settings are restricting your access to the <code>myproxy-server</code>. It is possible that the <code>myproxy-server</code> is running with the default <code>myproxy-server.config</code> file, which does not authorize any operations.</p>	<p>See Configuring for more information.</p>

Glossary

H

host certificate

An EEC belonging to a host. When using GSI this certificate is typically stored in `/etc/grid-security/hostcert.pem`. For more information on possible host certificate locations see the [GSI C Developer's Guide](#).

DRAFT