

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 bind(): Address already in use	<p>This error indicates that the myproxy-server port (default: 7512) is in use by another process, probably another myproxy-server instance. You cannot run multiple instances of the myproxy-server on the same network port.</p>	<p>If you want to run multiple instances of the myproxy-server 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 myproxy-server on that port.</p>
grid-proxy-init failed	<p>This error indicates that the grid-proxy-init command failed when myproxy-init attempted to run it, which implies a problem with the underlying Globus installation.</p>	<p>Run <code>grid-proxy-init -debug -verify</code> for more information.</p>
User not authorized	<p>An error from the myproxy-server 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 myproxy-server. It is possible that the myproxy-server 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

GT 4.2.0 MyProxy: User's Guide

DRAFT

GT 4.2.0 MyProxy: User's Guide

DRAFT

Table of Contents

1. Introduction	1
2. Using MyProxy	2
1. Storing a credential in the MyProxy repository	2
2. Retrieving a credential from the MyProxy repository	2
I. Command-line tools	3
myproxy-init	4
myproxy-info	8
myproxy-logon	9
myproxy-store	11
myproxy-retrieve	15
myproxy-destroy	17
myproxy-change-pass-phrase	18
myproxy-admin-adduser	19
myproxy-admin-change-pass	21
myproxy-admin-query	22
myproxy-admin-load-credential	23
myproxy-server	27
3. Debugging	28
1. Debugging GSI authentication problems	28
4. Troubleshooting	29
1. Incorrect system clocks	29
2. Errors	30
Glossary	31

List of Tables

1. myproxy-init options	6
2. myproxy-info options	8
3. myproxy-logon options	10
4. myproxy-store options	13
5. myproxy-retrieve options	16
6. myproxy-destroy options	17
7. myproxy-change-pass-phrase options	18
8. myproxy-admin-adduser options	20
9. myproxy-admin-change-pass options	21
10. myproxy-admin-query options	22
11. myproxy-admin-load-credential options	25
12. myproxy-server options	27
4.1. MyProxy Errors	30

DRAFT

Chapter 1. Introduction

The MyProxy User's Guide provides general end user-oriented information. The major end-user issues are storing and retrieving credentials in the MyProxy repository.

DRAFT

Chapter 2. Using MyProxy

1. Storing a credential in the MyProxy repository

Rather than storing your X.509 credentials (certificate and *private key*) on each machine you use, you can store them in a MyProxy repository and retrieve a *proxy credential* from the MyProxy repository when needed.

To store a credential in the MyProxy repository, run the **myproxy-init** command on a computer where your Grid credentials are located. For example:

```
$ myproxy-init -a -s myproxy.ncsa.uiuc.edu
Your identity: /C=US/O=National Computational Science Alliance/CN=Jim Basney
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Your proxy is valid until Fri Sep 13 13:52:56 2002
Enter MyProxy Pass Phrase:
Verifying password - Enter MyProxy Pass Phrase:
A proxy valid for 168 hours (7.0 days) for user jbasney now exists on myproxy.ncsa
```

The **myproxy-init** command prompts first for the pass phrase of your private key (similar to **grid-proxy-init**) and then prompts twice for a new pass phrase to use to secure the credentials on the MyProxy server. By default, the credential is stored under your Unix username (jbasney in the example above) for 7 days and can be used to retrieve credentials with 12 hour lifetimes. [Command-line tools](#) below lists all the available options for the myproxy-init command.

2. Retrieving a credential from the MyProxy repository

Once you've stored a credential in the MyProxy repository, you can retrieve a proxy credential whenever you need one with the **myproxy-logon** command. For example:

```
$ myproxy-logon -s myproxy.ncsa.uiuc.edu
Enter MyProxy Pass Phrase:
A proxy has been received for user jbasney in /tmp/x509up_u500
```

The **myproxy-logon** command prompts for the pass phrase you set previously with **myproxy-init**, retrieves a proxy credential for you, and stores it in the correct default location for use with other Globus Toolkit programs. The [Command-line tools](#) lists all the available options for the **myproxy-logon** command.

Command-line tools

DRAFT

Name

myproxy-init -- Store a *proxy credential* for later retrieval

myproxy-init

Tool description

The **myproxy-init** command uploads a credential to a **myproxy-server** for later retrieval. In the default mode, the command first prompts for the user's Grid pass phrase (if needed), which is used to create a proxy credential. The command then prompts for a MyProxy pass phrase, which will be required to later retrieve the credential. The MyProxy pass phrase must be entered a second time for confirmation. A credential with a lifetime of one week (by default) is then delegated to the **myproxy-server** and stored with the given MyProxy pass phrase. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon** using the MyProxy passphrase. The default behavior can be overridden by options specified below.

The **myproxy-init** command can also upload a credential to a **myproxy-server** to support credential renewal. Renewal allows a trusted service (for example, a batch job scheduler) to obtain a new credential for a user before the existing credential it has for that user expires. The **-R** argument to **myproxy-init** configures the credential for renewal by the specified service. Renewal requires two authentications. The renewing service must authenticate with its own credentials, matching the distinguished name specified by the **-R** argument, and must also authenticate with an existing credential that matches the distinguished name of the stored credential to retrieve a new credential.

A credential may be used either for retrieval or renewal, but not both. If both are desired, upload a different credential for each use with a different name, using the **-k** option.

The hostname where the **myproxy-server** is running must be specified by either defining the **MYPROXY_SERVER** environment variable or the **-s** option.

By default, **myproxy-init** will create a proxy credential from the user's end-entity credentials at `~/.globus/usercert.pem` and `~/.globus/userkey.pem` to delegate to the **myproxy-server**. To specify an alternate location for the source certificate and key to delegate, use the **X509_USER_CERT** and **X509_USER_KEY** environment variables. To use a proxy credential as the source of the delegation, set both environment variables to the location of the proxy credential. To delegate a "legacy globus proxy", set the **GT_PROXY_MODE** environment variable to "old".

Command syntax

myproxy-init [options]

Command options

DRAFT

Table 1. myproxy-init options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-P, --pidfile <i>path</i>	Specifies a file to write the pid to.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-c <i>hours</i> , --cred_lifetime <i>hours</i>	Specifies the lifetime of the credential stored on the myproxy-server in hours. Specify 0 for the maximum possible lifetime, i.e., the lifetime of the original credential. Default: 1 week (168 hours).
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-d, --dn_as_username	Use the <i>certificate subject</i> (DN) as the default username, instead of the LOGNAME environment variable.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. By default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for the sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.

-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-K <i>description</i>	blank
--creddesc <i>description</i>	Specifies credential description.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

DRAFT

Name

myproxy-info -- Display information about credentials

myproxy-info

Tool description

The **myproxy-info** command displays information about a user's credentials stored on a **myproxy-server**. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logon** when running this command.

Command syntax

myproxy-info [options]

Command options

Table 2. myproxy-info options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l <i>name</i> , --username <i>name</i>	Specifies the MyProxy account to query. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.

Name

myproxy-logon -- Retrieve a credential

myproxy-logon

Tool description

The **myproxy-logon** command retrieves a credential from the **myproxy-server** that was previously stored using **myproxy-init**. In the default mode, the command prompts for the MyProxy pass phrase associated with the credential to be retrieved and stores the retrieved credential in the standard location (*/tmp/x509up_u<uid>*).

If the repository contains an end-entity certificate, this command will retrieve an RFC 3820 compliant proxy (also known as "proxy draft compliant impersonation proxy") by default. Set the the GT_PROXY_MODE environment variable to "old" to retrieve a "legacy globus proxy" instead. If the repository contains a *proxy certificate*, the retrieved proxy will always be of the same type as the stored proxy.

The **myproxy-logon** is also available under the name **myproxy-get-delegation** for backward compatibility.

Command syntax

myproxy-logon [options]

Command options

Table 3. myproxy-logon options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to retrieve is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable. When used with the -a option, the certificate subject of the authorization credential is used. Otherwise, the certificate subject of the default credential is used.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the lifetime of credentials retrieved from the myproxy-server using the stored credential. The resulting lifetime is the shorter of the requested lifetime and the lifetime specified when the credential was stored using myproxy-init . Default: 12 hours.
-o <i>file</i> , --out <i>file</i>	Specifies where the retrieved proxy credential should be stored. If this option is not specified, the proxy credential will be stored in the default location (/tmp/x509up_u<uid>).
-a <i>file</i> , --authorization <i>file</i>	Specifies a credential to be used for authorizing the request instead of a passphrase. When renewing a credential, use this option to specify the existing, valid credential that you want to renew. Renewing a credential generally requires two certificate-based authentications. The client authenticates with its identity, using the credential in the standard location or specified by X509_USER_PROXY or X509_USER_CERT and X509_USER_KEY in addition to authenticating with the existing credential, in the location specified by this option, that it wants to renew.
-k <i>name</i> , --credname <i>name</i>	Specifies the name of the credential that is to be retrieved or renewed.
-S, --stdin_pass	By default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

myproxy-store -- Store end-entity credential for later retrieval

myproxy-store

Tool description

The **myproxy-store** command uploads a credential to a **myproxy-server(8)** for later retrieval. Unlike **myproxy-init(1)**, this command transfers the private key over the network (over a private channel). In the default mode, the command will take the credentials found in `~/.globus/usercert.pem` and `~/.globus/userkey.pem` and store them in the **myproxy-server(8)** repository. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon(1)** using the credential passphrase. The default behavior can be overridden by options specified below.

The hostname where the **myproxy-server(8)** is running must be specified by either defining the `MYPROXY_SERVER` environment variable or the `-s` option.

Command syntax

myproxy-store [options]

Command options

DRAFT

Table 4. myproxy-store options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server(8) . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of the source certificate. This is a required parameter.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of the source private key. This is a required parameter.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server(8) using the stored credential. Default: 12 hours
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config(5) default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config(5) server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-E <i>dn</i> , --retrieve_key <i>dn</i>	Allow the specified entity to retrieve end-entity credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.

-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-K <i>description</i> , --creddesc <i>description</i>	Specifies credential description.

Name

myproxy-retrieve -- Retrieve an end-entity credential

myproxy-retrieve

Tool description

The **myproxy-retrieve** command retrieves a credential directly from the **myproxy-server(8)** that was previously stored using **myproxy-init(1)** or **myproxy-store(1)**. Unlike **myproxy-logon(1)**, this command transfers the *private key* in the repository over the network (over a private channel). To obtain a proxy credential, we recommend using **myproxy-logon(1)** instead.

In the default mode, the command prompts for the pass phrase associated with the credential to be retrieved and stores the retrieved credential in the standard location (`~/.globus/usercert.pem` and `~/.globus/userkey.pem`). You could then run **grid-proxy-init** to create a proxy credential from the retrieved credentials.

Command syntax

myproxy-retrieve [options]

Command options

Table 5. myproxy-retrieve options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server(8) . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to retrieve is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable. When used with the -a option, the certificate subject of the authorization credential is used. Otherwise, the certificate subject of the default credential is used.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the lifetime of credentials retrieved from the myproxy-server(8) using the stored credential. The resulting lifetime is the shorter of the requested lifetime and the lifetime specified when the credential was stored using myproxy-init(1) . Default: 12 hours.
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of where the certificate will be stored.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of where the private key will be stored.
-a <i>file</i> , --authorization <i>file</i>	Specifies a credential to be used for authorizing the request instead of a passphrase. When renewing a credential, use this option to specify the existing, valid credential that you want to renew. Renewing a credential generally requires two certificate-based authentications. The client authenticates with its identity, using the credential in the standard location or specified by X509_USER_PROXY or X509_USER_CERT and X509_USER_KEY in addition to authenticating with the existing credential, in the location specified by this option, that it wants to renew.
-k <i>name</i> , --credname <i>name</i>	Specifies the name of the credential that is to be retrieved or renewed.
-S, --stdin_pass	By default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non- interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

myproxy-destroy -- Remove a credential from the repository

myproxy-destroy

Tool description

The **myproxy-destroy** command removes a credential from the **myproxy-server** that was previously stored using **myproxy-init**. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logout** when running this command.

Command syntax

myproxy-destroy [options]

Command options

Table 6. myproxy-destroy options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to destroy is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-k <i>name</i> , --credname <i>name</i>	Specifies name of the credential to be destroyed.

Name

myproxy-change-pass-phrase -- Change a credential's passphrase

myproxy-change-pass-phrase

Tool description

The **myproxy-change-pass-phrase** command changes the passphrase under which a credential is protected in the MyProxy repository. The command first prompts for the current passphrase for the credential, then prompts twice for the new passphrase. Only the credential owner can change a credential's passphrase. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logon** when running this command.

Command syntax

myproxy-change-pass-phrase [options]

Command options

Table 7. myproxy-change-pass-phrase options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command noninteractively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

`myproxy-admin-adduser` -- Add a new *user credential*

`myproxy-admin-adduser`

Tool description

The **myproxy-admin-adduser** command creates a new credential for a user and loads it into the MyProxy repository. It is a **perl** script that runs **grid-cert-request** (a standard Globus Toolkit program) and **grid-ca-sign** (from the Globus Simple CA package) to create the credential and then runs **myproxy-admin-load-credential** to load the credential into the MyProxy repository. The command prompts for the common name to be included in the new certificate (if the **-c** argument is not specified), the Globus Simple CA key password for signing the certificate, the MyProxy username (if the **-l** or **-d** arguments are not specified), and the MyProxy passphrase for the credential. Most of the command-line options for this command are passed directly to the **myproxy-admin-load-credential** command. The Globus Simple CA must be configured before using this command.

Command syntax

`myproxy-admin-adduser [options]`

Command options

Table 8. myproxy-admin-adduser options

-h	Displays command usage text and exits.
-u	Displays command usage text and exits.
-c <i>cn</i>	Specifies the Common Name for the new credential (for example: "Jim Basney").
-s <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.
-l <i>username</i>	Specifies the MyProxy account under which the credential should be stored.
-t <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-n	Disables passphrase authentication for the stored credential. If specified, the command will not prompt for a passphrase, the credential will not be encrypted by a passphrase in the repository, and the credential will not be retrievable using passphrase authentication with myproxy-logon . This option is used for storing renewable credentials and is implied by -R .
-d	Use the certificate subject (DN) as the username.
-a	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i>	Specifies the credential name.
-K <i>description</i>	Specifies credential description.

Name

myproxy-admin-change-pass -- Change credential passphrase

myproxy-admin-change-pass

Tool description

The **myproxy-admin-change-pass** command changes the passphrase used to encrypt a credential in the MyProxy repository. The command first prompts for the current passphrase for the credential, then prompts twice for the new passphrase. If an empty passphrase is given, the credential will not be encrypted. It accesses the repository directly and must be run on the machine where the **myproxy-server** is installed from the account that owns the repository.

Command syntax

myproxy-admin-change-pass [options]

Command options

Table 9. myproxy-admin-change-pass options

-h	Displays command usage text and exits.
-u	Displays command usage text and exits.
-s <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.
-l <i>username</i>	Specifies the MyProxy account under which the credential should be stored.
-k <i>name</i>	Specifies the credential name.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.

Name

myproxy-admin-query -- Query repository contents

myproxy-admin-query

Tool description

The **myproxy-admin-query** command displays information about the credentials stored in the MyProxy repository. It can also be used to remove credentials from the repository. It accesses the repository directly and must be run on the machine where the **myproxy-server** is installed from the account that owns the repository.

Command syntax

myproxy-admin-query [options]

Command options

Table 10. myproxy-admin-query options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-l <i>name</i> , --username <i>name</i>	Returns information on credentials for a single username. By default, the command returns information on all credentials for all usernames.
-k <i>name</i> , --credname <i>name</i>	Returns information on the credentials with the specified name.
-e <i>hours</i> , --expiring_in <i>hours</i>	Returns information on credentials with remaining lifetime less than the specified number of hours. For example, -e 0 will return all expired credentials.
-t <i>hours</i> , --time_left <i>hours</i>	Returns information on credentials with remaining lifetime greater than the specified number of hours.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCA-TION/var/myproxy.
-r, --remove	Remove the credentials matching the query from the repository. For example, <i>myproxy-admin-query -e 0 -r</i> will remove all expired credentials from the repository.
-L ' <i>msg</i> ', --lock ' <i>msg</i> '	Places the credentials matching the query under an administrative lock and specifies a message to be returned on access attempts. Be sure to put the message in quotes so it is captured as one argument to the command.
-U, --unlock	Removes any administrative locks for the credentials matching the query.

Name

`myproxy-admin-load-credential --` Directly load repository

`myproxy-admin-load-credential`

Tool description

The **myproxy-admin-load-credential** command stores a credential directly in the local MyProxy repository. It must be run from the account that owns the repository. Many of the options are similar to **myproxy-init**. However, unlike **myproxy-init**, **myproxy-admin-load-credential** does not create a proxy from the source credential but instead directly loads a copy of the source credential into the repository. The pass phrase of the source credential is unchanged. Use **myproxy-admin-change-pass** to change the pass phrase after the credential is stored if desired. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon** using the MyProxy passphrase. The command's behavior is controlled by the following options.

Command syntax

`myproxy-admin-load-credential [options]`

DRAFT

Command options

DRAFT

Table 11. myproxy-admin-load-credential options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of the source certificate. This is a required parameter.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of the source private key. This is a required parameter.
-l <i>username</i> , --username <i>user-name</i>	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-d, --dn_as_username	Use the certificate subject (DN) as the username.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.

<i>-K description, --creddesc de- scription</i>	Specifies credential description.
---	-----------------------------------

DRAFT

Name

myproxy-server -- Store credentials in an online repository

myproxy-server

Tool description

The **myproxy-server** is a server that runs on a trusted, secure host and manages a database of security credentials for use from remote sites. The **myproxy-init** program stores credentials with associated policies that specify credential lifetimes and who is authorized to retrieve credentials. The **myproxy-server.config** file sets server-wide policies that are used in conjunction with the policies set by **myproxy-init** to control who is authorized to store and retrieve credentials.

Command syntax

myproxy-server [options]

Command options

Table 12. myproxy-server options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-d, --debug	Run the server in debug mode. In this mode, the server will run in the foreground, will accept one connection, write log messages to the terminal while processing the incoming request, and exit after completing one request.
-p <i>port</i> , --port <i>port</i>	Specifies the TCP port number that the myproxy-server should listen on. Default: 7512.
-c <i>file</i> , --config <i>file</i>	Specifies the location of the myproxy-server configuration file. Default: /etc/myproxy-server.config or \$GLOBUS_LOCATION/etc/myproxy-server.config.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.

Chapter 3. Debugging

In addition to the following, there is more debugging information in [Chapter 4, Testing](#) and [Chapter 6, Debugging](#) in the MyProxy Admin Guide.

1. Debugging GSI authentication problems

To debug GSI authentication problems, run

```
grid-proxy-init -debug -verify
```

from the terminal where you run the MyProxy clients, and run

```
grid-proxy-init -debug -verify -cert /etc/grid-security/hostcert.pem -key /etc/grid-security/hostkey.pem
```

as root on the myproxy-server machine (assuming you run the myproxy-server as root).

DRAFT

Chapter 4. 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 4.1. 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 passphrases 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>
grid-proxy-init 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 <code>grid-proxy-init -debug -verify</code> 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

C

certificate subject An identifier for the certificate owner, e.g. `"/DC=org/DC=doegrids/OU=People/CN=John Doe 123456"`. The subject is part of the information the CA binds to a public key when creating a certificate.

P

private key The private part of a key pair. Depending on the type of certificate the key corresponds to it may typically be found in `$HOME/.globus/userkey.pem` (for user certificates), `/etc/grid-security/hostkey.pem` (for host certificates) or `/etc/grid-security/<service>/<service>key.pem` (for service certificates).

For more information on possible private key locations see [this](#).

proxy certificate A short lived certificate issued using a EEC. A proxy certificate typically has the same effective subject as the EEC that issued it and can thus be used in its place. GSI uses proxy certificates for single sign on and delegation of rights to other entities.

For more information about types of proxy certificates and their compatibility in different versions of GT, see <http://dev.globus.org/wiki/Security/ProxyCertTypes>.

proxy credentials The combination of a proxy certificate and its corresponding private key. GSI typically stores proxy credentials in `/tmp/x509up_u<uid>` , where `<uid>` is the user id of the proxy owner.

U

user credentials The combination of a user certificate and its corresponding private key.

GT 4.2.0 MyProxy: Developer's Guide

DRAFT

GT 4.2.0 MyProxy: Developer's Guide

Introduction

We recommend using the [CoG Kits](http://www.cogkit.org/)¹ when developing with MyProxy.

DRAFT

¹ <http://www.cogkit.org/>

Table of Contents

1. Before you begin	1
1. Feature summary	1
2. Tested platforms	1
3. Backward compatibility summary	1
4. Technology dependencies	2
5. MyProxy Security Considerations	2
2. Usage scenarios	3
3. Tutorials	4
4. Architecture and design overview	5
5. APIs	6
6. Configuring	7
I. Command-line tools	10
myproxy-init	11
myproxy-info	15
myproxy-logon	16
myproxy-store	18
myproxy-retrieve	22
myproxy-destroy	24
myproxy-change-pass-phrase	25
myproxy-admin-adduser	26
myproxy-admin-change-pass	28
myproxy-admin-query	29
myproxy-admin-load-credential	30
myproxy-server	34
7. Environment variable interface	35
1. Environmental variables for MyProxy	35
8. Debugging	37
1. Debugging GSI authentication problems	37
9. Troubleshooting	38
1. Incorrect system clocks	38
2. Errors	39
10. Related Documentation	40
Glossary	41

List of Tables

6.1. myproxy-server.config lines	8
2. myproxy-init options	13
3. myproxy-info options	15
4. myproxy-logon options	17
5. myproxy-store options	20
6. myproxy-retrieve options	23
7. myproxy-destroy options	24
8. myproxy-change-pass-phrase options	25
9. myproxy-admin-adduser options	27
10. myproxy-admin-change-pass options	28
11. myproxy-admin-query options	29
12. myproxy-admin-load-credential options	32
13. myproxy-server options	34
7.1. Environment variables	36
9.1. MyProxy Errors	39

DRAFT

Chapter 1. Before you begin

1. Feature summary

Supported Features

- Users can store and retrieve multiple X.509 *proxy credentials* using **myproxy-init** and **myproxy-logon**.
- Users can store and retrieve multiple X.509 end-entity credentials using **myproxy-store** and **myproxy-retrieve**.
- Administrators can load the repository with X.509 end-entity credentials on the users' behalf using **myproxy-admin-load-credential**.
- Administrators can use the **myproxy-admin-adduser** command to create *user credentials* and load them into the MyProxy repository.
- Users and administrators can set access control policies on the credentials in the repository.
- If allowed by policy, job managers (such as Condor-G) can renew credentials before they expire.
- The MyProxy server enforces local site passphrase policies using a configurable external call-out.

Deprecated Features

- None

2. Tested platforms

Tested Platforms for MyProxy

- Mac OS X 10.3
- i686 GNU/Linux
- ia64 GNU/Linux

3. Backward compatibility summary

Protocol changes since GT 4.0.x

- None.

API changes since GT 4.0.x

- None.

Exception changes since GT 4.0.x

- Not applicable

Schema changes since GT 4.0.x

- Not applicable

4. Technology dependencies

MyProxy depends on the following GT component:

- Non-WS Authentication and Authorization

MyProxy depends on the following 3rd party software:

- None

5. 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.

Chapter 2. Usage scenarios

MyProxy provides a solution for delegating credentials to Grid portals to allow the portal to authenticate to Grid services on the user's behalf. A Grid portal is a web server that provides an interface to Grid services, allowing users to submit compute jobs, transfer files, and query Grid information services from a standard web browser. For example:

- The Open Grid Computing Environment¹ (OGCE) collaboratory provides software for building grid computing portals using MyProxy.
- The GridSphere² credential manager portlet supports MyProxy.
- The Grid Portal Toolkit³ interfaces with MyProxy using the Perl CoG Kit⁴.
- The Extreme! Computing Lab⁵'s Proxy Manager Xportlet⁶ interfaces with MyProxy.

¹ <http://www.ogce.org/>

² <http://www.gridsphere.org/>

³ <https://gridport.npaci.edu/>

⁴ <https://gridport.npaci.edu/cog/>

⁵ <http://www.extreme.indiana.edu/>

⁶ <http://www.extreme.indiana.edu/xportlets/project/index.shtml>

Chapter 3. Tutorials

There are no tutorials available at this time.

DRAFT

Chapter 4. Architecture and design overview

The MyProxy system architecture and design is described in the following two publications:

- J. Basney, M. Humphrey, and V. Welch. The MyProxy Online Credential Repository¹. Software: Practice and Experience, 2005.
- J. Novotny, S. Tuecke, and V. Welch. An Online Credential Repository for the Grid: MyProxy². Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10), IEEE Press, August 2001.

DRAFT

¹ <http://www.ncsa.uiuc.edu/~jbasney/myproxy-spe.pdf>

² <http://www.globus.org/alliance/publications/papers/myproxy.pdf>

Chapter 5. APIs

A [Java API](#)¹ is available.

DRAFT

¹ <http://www.globus.org/cog/distribution/1.2/api/org/globus/myproxy/package-summary.html>

Chapter 6. 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 6.1. 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

Command-line tools

DRAFT

Name

myproxy-init -- Store a *proxy credential* for later retrieval

myproxy-init

Tool description

The **myproxy-init** command uploads a credential to a **myproxy-server** for later retrieval. In the default mode, the command first prompts for the user's Grid pass phrase (if needed), which is used to create a proxy credential. The command then prompts for a MyProxy pass phrase, which will be required to later retrieve the credential. The MyProxy pass phrase must be entered a second time for confirmation. A credential with a lifetime of one week (by default) is then delegated to the **myproxy-server** and stored with the given MyProxy pass phrase. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon** using the MyProxy passphrase. The default behavior can be overridden by options specified below.

The **myproxy-init** command can also upload a credential to a **myproxy-server** to support credential renewal. Renewal allows a trusted service (for example, a batch job scheduler) to obtain a new credential for a user before the existing credential it has for that user expires. The **-R** argument to **myproxy-init** configures the credential for renewal by the specified service. Renewal requires two authentications. The renewing service must authenticate with its own credentials, matching the distinguished name specified by the **-R** argument, and must also authenticate with an existing credential that matches the distinguished name of the stored credential to retrieve a new credential.

A credential may be used either for retrieval or renewal, but not both. If both are desired, upload a different credential for each use with a different name, using the **-k** option.

The hostname where the **myproxy-server** is running must be specified by either defining the **MYPROXY_SERVER** environment variable or the **-s** option.

By default, **myproxy-init** will create a proxy credential from the user's end-entity credentials at `~/globus/usercert.pem` and `~/globus/userkey.pem` to delegate to the **myproxy-server**. To specify an alternate location for the source certificate and key to delegate, use the **X509_USER_CERT** and **X509_USER_KEY** environment variables. To use a proxy credential as the source of the delegation, set both environment variables to the location of the proxy credential. To delegate a "legacy globus proxy", set the **GT_PROXY_MODE** environment variable to "old".

Command syntax

myproxy-init [options]

Command options

DRAFT

Table 2. myproxy-init options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-P, --pidfile <i>path</i>	Specifies a file to write the pid to.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-c <i>hours</i> , --cred_lifetime <i>hours</i>	Specifies the lifetime of the credential stored on the myproxy-server in hours. Specify 0 for the maximum possible lifetime, i.e., the lifetime of the original credential. Default: 1 week (168 hours).
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-d, --dn_as_username	Use the <i>certificate subject</i> (DN) as the default username, instead of the LOGNAME environment variable.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. By default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for the sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.

-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-K <i>description</i>	blank
--creddesc <i>description</i>	Specifies credential description.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

DRAFT

Name

myproxy-info -- Display information about credentials

myproxy-info

Tool description

The **myproxy-info** command displays information about a user's credentials stored on a **myproxy-server**. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logon** when running this command.

Command syntax

myproxy-info [options]

Command options

Table 3. myproxy-info options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l <i>name</i> , --username <i>name</i>	Specifies the MyProxy account to query. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.

Name

myproxy-logon -- Retrieve a credential

myproxy-logon

Tool description

The **myproxy-logon** command retrieves a credential from the **myproxy-server** that was previously stored using **myproxy-init**. In the default mode, the command prompts for the MyProxy pass phrase associated with the credential to be retrieved and stores the retrieved credential in the standard location (*/tmp/x509up_u<uid>*).

If the repository contains an end-entity certificate, this command will retrieve an RFC 3820 compliant proxy (also known as "proxy draft compliant impersonation proxy") by default. Set the the GT_PROXY_MODE environment variable to "old" to retrieve a "legacy globus proxy" instead. If the repository contains a *proxy certificate*, the retrieved proxy will always be of the same type as the stored proxy.

The **myproxy-logon** is also available under the name **myproxy-get-delegation** for backward compatibility.

Command syntax

myproxy-logon [options]

Command options

Table 4. myproxy-logon options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to retrieve is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable. When used with the -a option, the certificate subject of the authorization credential is used. Otherwise, the certificate subject of the default credential is used.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the lifetime of credentials retrieved from the myproxy-server using the stored credential. The resulting lifetime is the shorter of the requested lifetime and the lifetime specified when the credential was stored using myproxy-init . Default: 12 hours.
-o <i>file</i> , --out <i>file</i>	Specifies where the retrieved proxy credential should be stored. If this option is not specified, the proxy credential will be stored in the default location (/tmp/x509up_u<uid>).
-a <i>file</i> , --authorization <i>file</i>	Specifies a credential to be used for authorizing the request instead of a passphrase. When renewing a credential, use this option to specify the existing, valid credential that you want to renew. Renewing a credential generally requires two certificate-based authentications. The client authenticates with its identity, using the credential in the standard location or specified by X509_USER_PROXY or X509_USER_CERT and X509_USER_KEY in addition to authenticating with the existing credential, in the location specified by this option, that it wants to renew.
-k <i>name</i> , --credname <i>name</i>	Specifies the name of the credential that is to be retrieved or renewed.
-S, --stdin_pass	By default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

myproxy-store -- Store end-entity credential for later retrieval

myproxy-store

Tool description

The **myproxy-store** command uploads a credential to a **myproxy-server(8)** for later retrieval. Unlike **myproxy-init(1)**, this command transfers the private key over the network (over a private channel). In the default mode, the command will take the credentials found in `~/.globus/usercert.pem` and `~/.globus/userkey.pem` and store them in the **myproxy-server(8)** repository. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon(1)** using the credential passphrase. The default behavior can be overridden by options specified below.

The hostname where the **myproxy-server(8)** is running must be specified by either defining the `MYPROXY_SERVER` environment variable or the `-s` option.

Command syntax

myproxy-store [options]

Command options

DRAFT

Table 5. myproxy-store options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server(8) . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of the source certificate. This is a required parameter.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of the source private key. This is a required parameter.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server(8) using the stored credential. Default: 12 hours
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config(5) default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config(5) server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-E <i>dn</i> , --retrieve_key <i>dn</i>	Allow the specified entity to retrieve end-entity credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.

-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. by default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-K <i>description</i> , --creddesc <i>description</i>	Specifies credential description.

Name

myproxy-retrieve -- Retrieve an end-entity credential

myproxy-retrieve

Tool description

The **myproxy-retrieve** command retrieves a credential directly from the **myproxy-server(8)** that was previously stored using **myproxy-init(1)** or **myproxy-store(1)**. Unlike **myproxy-logon(1)**, this command transfers the *private key* in the repository over the network (over a private channel). To obtain a proxy credential, we recommend using **myproxy-logon(1)** instead.

In the default mode, the command prompts for the pass phrase associated with the credential to be retrieved and stores the retrieved credential in the standard location (`~/.globus/usercert.pem` and `~/.globus/userkey.pem`). You could then run **grid-proxy-init** to create a proxy credential from the retrieved credentials.

Command syntax

myproxy-retrieve [options]

Command options

Table 6. myproxy-retrieve options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server(8) . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to retrieve is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable. When used with the -a option, the certificate subject of the authorization credential is used. Otherwise, the certificate subject of the default credential is used.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the lifetime of credentials retrieved from the myproxy-server(8) using the stored credential. The resulting lifetime is the shorter of the requested lifetime and the lifetime specified when the credential was stored using myproxy-init(1) . Default: 12 hours.
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of where the certificate will be stored.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of where the private key will be stored.
-a <i>file</i> , --authorization <i>file</i>	Specifies a credential to be used for authorizing the request instead of a passphrase. When renewing a credential, use this option to specify the existing, valid credential that you want to renew. Renewing a credential generally requires two certificate-based authentications. The client authenticates with its identity, using the credential in the standard location or specified by X509_USER_PROXY or X509_USER_CERT and X509_USER_KEY in addition to authenticating with the existing credential, in the location specified by this option, that it wants to renew.
-k <i>name</i> , --credname <i>name</i>	Specifies the name of the credential that is to be retrieved or renewed.
-S, --stdin_pass	By default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non- interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

myproxy-destroy -- Remove a credential from the repository

myproxy-destroy

Tool description

The **myproxy-destroy** command removes a credential from the **myproxy-server** that was previously stored using **myproxy-init**. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logout** when running this command.

Command syntax

myproxy-destroy [options]

Command options

Table 7. myproxy-destroy options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential to destroy is stored. By default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-k <i>name</i> , --credname <i>name</i>	Specifies name of the credential to be destroyed.

Name

myproxy-change-pass-phrase -- Change a credential's passphrase

myproxy-change-pass-phrase

Tool description

The **myproxy-change-pass-phrase** command changes the passphrase under which a credential is protected in the MyProxy repository. The command first prompts for the current passphrase for the credential, then prompts twice for the new passphrase. Only the credential owner can change a credential's passphrase. The user must have a valid proxy credential as generated by **grid-proxy-init** or retrieved by **myproxy-logon** when running this command.

Command syntax

myproxy-change-pass-phrase [options]

Command options

Table 8. myproxy-change-pass-phrase options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>hostname</i> , --pshost <i>hostname</i>	Specifies the hostname of the myproxy-server. This option is required if the MYPROXY_SERVER environment variable is not defined. If specified, this option overrides the MYPROXY_SERVER environment variable.
-p <i>port</i> , --psport <i>port</i>	Specifies the TCP port number of the myproxy-server . Default: 7512.
-l, --username	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-d, --dn_as_username	Use the certificate subject (DN) as the default username, instead of the LOGNAME environment variable.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command noninteractively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.

Name

`myproxy-admin-adduser` -- Add a new *user credential*

`myproxy-admin-adduser`

Tool description

The **myproxy-admin-adduser** command creates a new credential for a user and loads it into the MyProxy repository. It is a **perl** script that runs **grid-cert-request** (a standard Globus Toolkit program) and **grid-ca-sign** (from the Globus Simple CA package) to create the credential and then runs **myproxy-admin-load-credential** to load the credential into the MyProxy repository. The command prompts for the common name to be included in the new certificate (if the **-c** argument is not specified), the Globus Simple CA key password for signing the certificate, the MyProxy username (if the **-l** or **-d** arguments are not specified), and the MyProxy passphrase for the credential. Most of the command-line options for this command are passed directly to the **myproxy-admin-load-credential** command. The Globus Simple CA must be configured before using this command.

Command syntax

`myproxy-admin-adduser [options]`

Command options

Table 9. myproxy-admin-adduser options

-h	Displays command usage text and exits.
-u	Displays command usage text and exits.
-c <i>cn</i>	Specifies the Common Name for the new credential (for example: "Jim Basney").
-s <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.
-l <i>username</i>	Specifies the MyProxy account under which the credential should be stored.
-t <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-n	Disables passphrase authentication for the stored credential. If specified, the command will not prompt for a passphrase, the credential will not be encrypted by a passphrase in the repository, and the credential will not be retrievable using passphrase authentication with myproxy-logon . This option is used for storing renewable credentials and is implied by -R .
-d	Use the certificate subject (DN) as the username.
-a	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead. This option implies -n since passphrase authentication is not used for credential renewal.
-x	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i>	Specifies the credential name.
-K <i>description</i>	Specifies credential description.

Name

myproxy-admin-change-pass -- Change credential passphrase

myproxy-admin-change-pass

Tool description

The **myproxy-admin-change-pass** command changes the passphrase used to encrypt a credential in the MyProxy repository. The command first prompts for the current passphrase for the credential, then prompts twice for the new passphrase. If an empty passphrase is given, the credential will not be encrypted. It accesses the repository directly and must be run on the machine where the **myproxy-server** is installed from the account that owns the repository.

Command syntax

myproxy-admin-change-pass [options]

Command options

Table 10. myproxy-admin-change-pass options

-h	Displays command usage text and exits.
-u	Displays command usage text and exits.
-s <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.
-l <i>username</i>	Specifies the MyProxy account under which the credential should be stored.
-k <i>name</i>	Specifies the credential name.
-S, --stdin_pass	by default, the command prompts for a passphrase and reads the passphrase from the active tty. When running the command non-interactively, there may be no associated tty. Specifying this option tells the command to read passphrases from standard input without prompts or confirmation.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.

Name

myproxy-admin-query -- Query repository contents

myproxy-admin-query

Tool description

The **myproxy-admin-query** command displays information about the credentials stored in the MyProxy repository. It can also be used to remove credentials from the repository. It accesses the repository directly and must be run on the machine where the **myproxy-server** is installed from the account that owns the repository.

Command syntax

myproxy-admin-query [options]

Command options

Table 11. myproxy-admin-query options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-l <i>name</i> , --username <i>name</i>	Returns information on credentials for a single username. By default, the command returns information on all credentials for all usernames.
-k <i>name</i> , --credname <i>name</i>	Returns information on the credentials with the specified name.
-e <i>hours</i> , --expiring_in <i>hours</i>	Returns information on credentials with remaining lifetime less than the specified number of hours. For example, -e 0 will return all expired credentials.
-t <i>hours</i> , --time_left <i>hours</i>	Returns information on credentials with remaining lifetime greater than the specified number of hours.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCA-TION/var/myproxy.
-r, --remove	Remove the credentials matching the query from the repository. For example, <i>myproxy-admin-query -e 0 -r</i> will remove all expired credentials from the repository.
-L ' <i>msg</i> ', --lock ' <i>msg</i> '	Places the credentials matching the query under an administrative lock and specifies a message to be returned on access attempts. Be sure to put the message in quotes so it is captured as one argument to the command.
-U, --unlock	Removes any administrative locks for the credentials matching the query.

Name

`myproxy-admin-load-credential --` Directly load repository

`myproxy-admin-load-credential`

Tool description

The **myproxy-admin-load-credential** command stores a credential directly in the local MyProxy repository. It must be run from the account that owns the repository. Many of the options are similar to **myproxy-init**. However, unlike **myproxy-init**, **myproxy-admin-load-credential** does not create a proxy from the source credential but instead directly loads a copy of the source credential into the repository. The pass phrase of the source credential is unchanged. Use **myproxy-admin-change-pass** to change the pass phrase after the credential is stored if desired. Proxy credentials with a default lifetime of 12 hours can then be retrieved by **myproxy-logon** using the MyProxy passphrase. The command's behavior is controlled by the following options.

Command syntax

`myproxy-admin-load-credential [options]`

DRAFT

Command options

DRAFT

Table 12. myproxy-admin-load-credential options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy
-c <i>filename</i> , --certfile <i>filename</i>	Specifies the filename of the source certificate. This is a required parameter.
-y <i>filename</i> , --keyfile <i>filename</i>	Specifies the filename of the source private key. This is a required parameter.
-l <i>username</i> , --username <i>user-name</i>	Specifies the MyProxy account under which the credential should be stored. by default, the command uses the value of the LOGNAME environment variable. Use this option to specify a different account username on the MyProxy server. The MyProxy username need not correspond to a real Unix username.
-t <i>hours</i> , --proxy_lifetime <i>hours</i>	Specifies the maximum lifetime of credentials retrieved from the myproxy-server using the stored credential. Default: 12 hours.
-d, --dn_as_username	Use the certificate subject (DN) as the username.
-a, --allow_anonymous_retrievers	Allow credentials to be retrieved with just pass phrase authentication. by default, only entities with credentials that match the myproxy-server.config default retriever policy may retrieve credentials. This option allows entities without existing credentials to retrieve a credential using pass phrase authentication by including "anonymous" in the set of allowed retrievers. The myproxy-server.config server-wide policy must also allow "anonymous" clients for this option to have an effect.
-A, --allow_anonymous_renewers	Allow credentials to be renewed by any client. Any client with a valid credential with a subject name that matches the stored credential may retrieve a new credential from the MyProxy repository if this option is given. Since this effectively defeats the purpose of proxy credential lifetimes, it is not recommended. It is included only for sake of completeness.
-r <i>dn</i> , --retrievable_by <i>dn</i>	Allow the specified entity to retrieve credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "Jim Basney"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=Jim Basney") instead.
-R <i>dn</i> , --renewable_by <i>dn</i>	Allow the specified entity to renew credentials. By default, the argument will be matched against the common name (CN) of the client (for example: "condorg/modi4.ncsa.uiuc.edu"). Specify -x before this option to match against the full distinguished name (DN) (for example: "/C=US/O=National Computational Science Alliance/CN=condorg/modi4.ncsa.uiuc.edu") instead.
-x, --regex_dn_match	Specifies that the DN used by options -r and -R will be matched as a regular expression.
-X, --match_cn_only	Specifies that the DN used by options -r and -R will be matched against the Common Name (CN) of the subject.
-k <i>name</i> , --credname <i>name</i>	Specifies the credential name.

<i>-K description, --creddesc de- scription</i>	Specifies credential description.
---	-----------------------------------

DRAFT

Name

myproxy-server -- Store credentials in an online repository

myproxy-server

Tool description

The **myproxy-server** is a server that runs on a trusted, secure host and manages a database of security credentials for use from remote sites. The **myproxy-init** program stores credentials with associated policies that specify credential lifetimes and who is authorized to retrieve credentials. The **myproxy-server.config** file sets server-wide policies that are used in conjunction with the policies set by **myproxy-init** to control who is authorized to store and retrieve credentials.

Command syntax

myproxy-server [options]

Command options

Table 13. myproxy-server options

-h, --help	Displays command usage text and exits.
-u, --usage	Displays command usage text and exits.
-v, --verbose	Enables verbose debugging output to the terminal.
-V, --version	Displays version information and exits.
-d, --debug	Run the server in debug mode. In this mode, the server will run in the foreground, will accept one connection, write log messages to the terminal while processing the incoming request, and exit after completing one request.
-p <i>port</i> , --port <i>port</i>	Specifies the TCP port number that the myproxy-server should listen on. Default: 7512.
-c <i>file</i> , --config <i>file</i>	Specifies the location of the myproxy-server configuration file. Default: /etc/myproxy-server.config or \$GLOBUS_LOCATION/etc/myproxy-server.config.
-s <i>dir</i> , --storage <i>dir</i>	Specifies the location of the credential storage directory. The directory must be accessible only by the user running the myproxy-server process for security reasons. Default: /var/myproxy or \$GLOBUS_LOCATION/var/myproxy.

Chapter 7. Environment variable interface

1. Environmental variables for MyProxy

DRAFT

Table 7.1. Environment variables

MYPROXY_SERVER	Specifies the hostname where the myproxy-server is running. This environment variable can be used in place of the <code>-s</code> option.
MYPROXY_SERVER_PORT	Specifies the port where the myproxy-server is running. This environment variable can be used in place of the <code>-p</code> option.
MYPROXY_SERVER_DN	Specifies the distinguished name (DN) of the myproxy-server . All MyProxy client programs authenticate the server's identity. By default, MyProxy servers run with host credentials, so the MyProxy client programs expect the server to have a distinguished name of the form "host/<fqhn>" or "myproxy/<fqhn>" (where <fqhn> is the fully-qualified hostname of the server). If the server is running with some other DN, you can set this environment variable to tell the MyProxy clients to accept the alternative DN.
X509_USER_CERT	Specifies a non-standard location for the certificate from which the <i>proxy credential</i> is created by myproxy-init . It also specifies an alternative location for the server's certificate. By default, the server uses <code>/etc/grid-security/hostcert.pem</code> when running as root or <code>~/ .globus/usercert.pem</code> when running as non-root.
X509_USER_KEY	Specifies a non-standard location for the <i>private key</i> from which the proxy credential is created by myproxy-init . It also specifies an alternative location for the server's private key. By default the server uses <code>/etc/grid-security/hostkey.pem</code> when running as root or <code>~/ .globus/userkey.pem</code> when running as non-root.
X509_USER_PROXY	Specifies an alternative location for the server's certificate and private key (in the same file). Use when running the server with a proxy credential. Note that the proxy will need to be periodically renewed before expiration to allow the myproxy-server to keep functioning. When the myproxy-server runs with a non-host credential, clients must have the MYPROXY_SERVER_DN environment variable set to the distinguished name of the certificate being used by the server.
GLOBUS_LOCATION	Specifies the root of the MyProxy installation, used to find the default location of the <code>myproxy-server.config</code> file and the credential storage directory.
LD_LIBRARY_PATH	The MyProxy server is typically linked dynamically with Globus security libraries, which must be present in the dynamic linker's search path. This typically requires <code>\$GLOBUS_LOCATION/lib</code> to be included in the list in the <code>LD_LIBRARY_PATH</code> environment variable, which is set by the <code>\$GLOBUS_LOCATION/libexec/globus-script-initializer</code> script, which should be called from any myproxy-server startup script. Alternatively, to set <code>LD_LIBRARY_PATH</code> appropriately for the Globus libraries in an interactive shell, source <code>\$GLOBUS_LOCATION/etc/globus-user-env.sh</code> (for sh shells) or <code>\$GLOBUS_LOCATION/etc/globus-user.env.csh</code> (for csh shells).
GT_PROXY_MODE	Set to "old" to use the "legacy globus proxy" format. By default, MyProxy uses the RFC 3820 compliant proxy (also known as "proxy draft compliant") format. If <code>GT_PROXY_MODE</code> is set to "old", then <code>myproxy-init</code> will store a legacy proxy and <code>myproxy-logon</code> will retrieve a legacy proxy (if possible). Note that if the repository contains a proxy certificate, rather than an end-entity certificate, the retrieved proxy will be of the same type as the stored proxy, regardless of the setting of this environment variable.

Chapter 8. Debugging

In addition to the following, there is more debugging information in [Chapter 4, Testing](#) and [Chapter 6, Debugging](#) in the MyProxy Admin Guide.

1. Debugging GSI authentication problems

To debug GSI authentication problems, run

```
grid-proxy-init -debug -verify
```

from the terminal where you run the MyProxy clients, and run

```
grid-proxy-init -debug -verify -cert /etc/grid-security/hostcert.pem -key /etc/grid-security/hostkey.pem
```

as root on the myproxy-server machine (assuming you run the myproxy-server as root).

DRAFT

Chapter 9. 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 9.1. 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>
grid-proxy-init 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 <code>grid-proxy-init -debug -verify</code> 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>

Chapter 10. Related Documentation

For additional information about MyProxy, see the [MyProxy Project Home Page](http://myproxy.ncsa.uiuc.edu/)¹ at NCSA.

DRAFT

¹ <http://myproxy.ncsa.uiuc.edu/>

Glossary

C

certificate subject An identifier for the certificate owner, e.g. `"/DC=org/DC=doegrids/OU=People/CN=John Doe 123456"`. The subject is part of the information the CA binds to a public key when creating a certificate.

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](#).

P

private key The private part of a key pair. Depending on the type of certificate the key corresponds to it may typically be found in `$HOME/.globus/userkey.pem` (for user certificates), `/etc/grid-security/hostkey.pem` (for host certificates) or `/etc/grid-security/<service>/<service>key.pem` (for service certificates).

For more information on possible private key locations see [this](#).

proxy certificate A short lived certificate issued using a EEC. A proxy certificate typically has the same effective subject as the EEC that issued it and can thus be used in its place. GSI uses proxy certificates for single sign on and delegation of rights to other entities.

For more information about types of proxy certificates and their compatibility in different versions of GT, see <http://dev.globus.org/wiki/Security/ProxyCertTypes>.

proxy credentials The combination of a proxy certificate and its corresponding private key. GSI typically stores proxy credentials in `/tmp/x509up_u<uid>`, where `<uid>` is the user id of the proxy owner.

U

user credentials The combination of a user certificate and its corresponding private key.

GT 4.2.0 Migrating Guide for MyProxy

Table of Contents

1. Migrating MyProxy from GT4.0	1
2. Migrating MyProxy from GT3	1
3. Migrating MyProxy from GT2	1

[Migrating Guide](#)

The following provides available information about migrating from previous versions of the Globus Toolkit.

1. Migrating MyProxy from GT4.0

No special procedures are required for MyProxy installations migrating from GT4.0 to GT4.2. MyProxy is backward compatible.

2. Migrating MyProxy from GT3

No special procedures are required for MyProxy installations migrating from GT3 to GT4. MyProxy is backward compatible.

3. Migrating MyProxy from GT2

No special procedures are required for MyProxy installations migrating from GT2 to GT4. MyProxy is backward compatible.

GT 4.2.0 MyProxy: Quality Profile

Table of Contents

1. Test coverage reports	1
2. Code analysis reports	1
3. Outstanding bugs	1
4. Bug Fixes	1
5. Performance reports	1

<titleabbrev>Quality Profile</titleabbrev>

1. Test coverage reports

Not yet available.

2. Code analysis reports

Not yet available.

3. Outstanding bugs

- [Bug 2709](#).¹ The MyProxy package isn't internationalized.

4. Bug Fixes

No changes have occurred for MyProxy.

5. Performance reports

- [MyProxy Scalability Information](#)²

¹ http://bugzilla.globus.org/globus/show_bug.cgi?id=2709

² <http://myproxy.ncsa.uiuc.edu/scalability.html>

GT 4.2.0 Release Notes: MyProxy

Table of Contents

1. Component Overview	1
2. Feature summary	1
3. Summary of Changes in MyProxy	2
4. Bug Fixes	2
5. Known Problems	2
6. Technology dependencies	2
7. Tested platforms	2
8. Backward compatibility summary	2
9. Associated Standards	3
10. For More Information	3
Glossary	3

<titleabbrev>Release Notes</titleabbrev>

1. Component Overview

MyProxy is an online credential repository. You can store X.509 *proxy credentials* in the MyProxy repository, protected by a passphrase, for later retrieval over the network. This eliminates the need for manually copying *private key* and certificate files between machines. MyProxy can also be used for authentication to grid portals and credential renewal with job managers.

2. Feature summary

Supported Features

- Users can store and retrieve multiple X.509 *proxy credentials* using **myproxy-init** and **myproxy-logon**.
- Users can store and retrieve multiple X.509 end-entity credentials using **myproxy-store** and **myproxy-retrieve**.
- Administrators can load the repository with X.509 end-entity credentials on the users' behalf using **myproxy-admin-load-credential**.
- Administrators can use the **myproxy-admin-adduser** command to create *user credentials* and load them into the MyProxy repository.
- Users and administrators can set access control policies on the credentials in the repository.
- If allowed by policy, job managers (such as Condor-G) can renew credentials before they expire.
- The MyProxy server enforces local site passphrase policies using a configurable external call-out.

Deprecated Features

- None

3. Summary of Changes in MyProxy

No changes have occurred for MyProxy since the last stable release, 4.0.x:

4. Bug Fixes

No changes have occurred for MyProxy.

5. Known Problems

The following problems and limitations are known to exist for MyProxy at the time of the 4.2.0 release:

5.1. Limitations

- No known limitations exist.

5.2. Outstanding bugs

- [Bug 2709](#).¹ The MyProxy package isn't internationalized.

6. Technology dependencies

MyProxy depends on the following GT component:

- Non-WS Authentication and Authorization

MyProxy depends on the following 3rd party software:

- None

7. Tested platforms

Tested Platforms for MyProxy

- Mac OS X 10.3
- i686 GNU/Linux
- ia64 GNU/Linux

8. Backward compatibility summary

Protocol changes since GT 4.0.x

- None.

API changes since GT 4.0.x

¹ http://bugzilla.globus.org/globus/show_bug.cgi?id=2709

- None.

Exception changes since GT 4.0.x

- Not applicable

Schema changes since GT 4.0.x

- Not applicable

9. Associated Standards

Associated standards for MyProxy:

- [GFD-E.054](http://www.ggf.org/documents/GFD.54.pdf)² MyProxy Protocol
- [RFC 3820](http://www.faqs.org/rfcs/rfc3820.html)³ Proxy Certificates
- [RFC 2246](http://www.faqs.org/rfcs/rfc2246.html)⁴ TLS

10. For More Information

See [MyProxy](#) for more information about this component.

Glossary

P

private key

The private part of a key pair. Depending on the type of certificate the key corresponds to it may typically be found in `$HOME/.globus/userkey.pem` (for user certificates), `/etc/grid-security/hostkey.pem` (for host certificates) or `/etc/grid-security/<service>/<service>key.pem` (for service certificates).

For more information on possible private key locations see [this](#).

proxy credentials

The combination of a proxy certificate and its corresponding private key. GSI typically stores proxy credentials in `/tmp/x509up_u<uid>`, where `<uid>` is the user id of the proxy owner.

U

user credentials

The combination of a user certificate and its corresponding private key.

² <http://www.ggf.org/documents/GFD.54.pdf>

³ <http://www.faqs.org/rfcs/rfc3820.html>

⁴ <http://www.faqs.org/rfcs/rfc2246.html>