

GT 4.2.0 GRAM2: Admin Guide

DRAFT

GT 4.2.0 GRAM2: Admin Guide

DRAFT

Table of Contents

| | |
|---|----|
| 1. Building and Installing GRAM | 1 |
| 2. Configuring GRAM | 2 |
| 1. Configuration Files | 2 |
| 2. Configure Inetd and Xinetd | 3 |
| 3. Advanced Configuration | 5 |
| 3. Job Manager | 6 |
| 1. Job Manager Setup | 6 |
| 2. Job Manager Configuration | 6 |
| 3. RSL Validation File Format | 6 |
| 4. Job Execution Environment | 6 |
| 5. RSL attributes | 7 |
| 6. Adding job managers | 7 |
| 4. Scheduler Event Generator / Job Manager Integration | 8 |
| 1. Introduction | 8 |
| 2. globus-job-manager-event-generator | 8 |
| 3. Job Manager Configuration | 9 |
| 4. globus-job-manager-event-generator Configuration | 10 |
| 5. Running the globus-job-manager-event-generator | 10 |
| 6. Troubleshooting the globus-job-manager-event-generator | 11 |
| 5. Audit Logging | 12 |
| 6. Testing GRAM | 13 |
| 7. Usage statistics collection by the Globus Alliance | 14 |

Chapter 1. Building and Installing GRAM

Gram will be installed during the normal installation of the GT4.2 and needs no extra steps.

DRAFT

Chapter 2. Configuring GRAM

1. Configuration Files

GRAM uses the following configuration files and directories:

1. [globus-gatekeeper.conf](#)¹
2. [globus-job-manager.conf](#)²
3. [grid-services](#)³
4. [/etc/grid-security/grid-mapfile](#)⁴

1. globus-gatekeeper.conf

Here is the default globus-gatekeeper.conf:

```
-x509_cert_dir /etc/grid-security/certificates
-x509_user_cert /etc/grid-security/hostcert.pem
-x509_user_key /etc/grid-security/hostkey.pem
-gridmap /etc/grid-security/grid-mapfile
-home /usr/local/globus
-e libexec
-logfile var/globus-gatekeeper.log
-port 2119
-grid_services etc/grid-services
-inetd
```

- *-x509_cert_dir* specifies where to find the trusted CA certificates.
- *-x509_user_cert* specifies where to find the gatekeeper cert.
- *-x509_user_key* specifies where to find the gatekeeper key.
- *-gridmap* specifies where to find the grid-mapfile.
- *-home* specifies where the *-e* and *-logfile* variables are relative to. By default, this is your `$GLOBUS_LOCATION`.
- *-e* specifies where to find scripts.
- *-logfile* specifies where the gatekeeper should put its log.
- *-port* specifies what port the gatekeeper will run on.
- *-grid_service* specifies where the directory which contains the configured jobmanagers is.
- *-inetd* specifies that the gatekeeper should exit after dealing with one request. That is because `inetd` will launch a copy of the gatekeeper for every request that comes in to the port in *-port*. If you are running a gatekeeper by hand, don't use this flag.

¹ #gram2-admin-configfile-gatekeeper

² #gram2-admin-configfile-jobmanager

³ #gram2-admin-configfile-gridservices

⁴ #gram2-admin-configfile-gridmapfile

2. globus-job-manager.conf

Here is an example globus-job-manager.conf:

```
-home "/home/bacon/pkgs/globus-2.4"
-globus-gatekeeper-host bacon.mcs.anl.gov
-globus-gatekeeper-port 2119
-globus-gatekeeper-subject "/O=Grid/O=Globus/CN=bacon.mcs.anl.gov"
-globus-host-cputype i686
-globus-host-manufacturer pc
-globus-host-osname Linux
-globus-host-osversion 2.2.19-4.7mdk
-save-logfile on_error
-state-file-dir /home/bacon/pkgs/globus-2.4/tmp
-machine-type unknown
```

See [Job Manager Configuration](#)⁵ for details. Note that the entries in this file are combined with the entries in \$GLOBUS_LOCATION/etc/grid-services for any specific jobmanager.

3. grid-services/

\$GLOBUS_LOCATION/etc/grid-services contains one file per configured jobmanager. The default jobmanager is contained in a file named "jobmanager". Actually this is a symbolic link to one of the jobmanager files located in the same directory that will be used as the default jobmanager. Here are the contents of an example file for a fork jobmanager:

```
stderr_log,local_cred - /home/bacon/pkgs/globus-2.4/libexec/globus-job-manager globus-job-
```

To install additional jobmanagers, you need to download the scheduler-specific jobmanager package from the [download page](#)⁶.

4. /etc/grid-security/grid-mapfile

The grid-mapfile specifies the list of authorized users of this resource. Each entry is a pairing of a subject name and a local user account. The location of this file is specified in globus-gatekeeper.conf

2. Configure Inetd and Xinetd

While running globus-personal-gatekeeper as a user is a good test, you will want to configure your machine to run globus-gatekeeper as root, so that other people will be able to use your gatekeeper. If you just run the personal gatekeeper, you won't have authority to su to other user accounts. To setup a full gatekeeper, you will need to make the following modifications as root:

In /etc/services, add the service name "gsigatekeeper" to port 2119.

```
gsigatekeeper      2119/tcp          # Globus Gatekeeper
```

Depending on whether your host is running inetd or xinetd, you will need to modify its configuration. If the directory /etc/xinetd.d/ exists, then your host is likely running xinetd. If the directory doesn't exist, your host is likely running inetd. Follow the appropriate instructions below according to what your host is running.

Inetd

⁵ #gram2-admin-jobmanager-config

⁶ <http://www.globus.org/toolkit/downloads/development/>

For inetd, add the following entry, all on one line, to /etc/inetd.conf. Be sure to replace GLOBUS_LOCATION below with the actual value of \$GLOBUS_LOCATION in your environment.

```
gsigatekeeper stream tcp nowait root
  /usr/bin/env env LD_LIBRARY_PATH=GLOBUS_LOCATION/lib
  GLOBUS_LOCATION/sbin/globus-gatekeeper
  -conf GLOBUS_LOCATION/etc/globus-gatekeeper.conf
```

This entry has changed from the entry provided for the gatekeeper in the Globus Toolkit 2.0 Administrator's Guide. The reason is that if you followed the instructions from the install section, you do not have a static gatekeeper. This requires you to set the LD_LIBRARY_PATH so that the gatekeeper can dynamically link against the libraries in \$GLOBUS_LOCATION/lib. To accomplish the setting of the environment variable in inetd, we use /usr/bin/env (the location may vary on your system) to first set LD_LIBRARY_PATH, and then to call the gatekeeper itself.

The advantage of this setup is that when you apply a security update to your installation, the gatekeeper will pick it up dynamically without your having to rebuild it.

Xinetd

For xinetd, add a file called "globus-gatekeeper" to the /etc/xinetd.d/ directory that has the following contents. Be sure to replace GLOBUS_LOCATION below with the actual value of \$GLOBUS_LOCATION in your environment.

```
service gsigatekeeper
{
  socket_type = stream
  protocol   = tcp
  wait       = no
  user       = root
  env        = LD_LIBRARY_PATH=GLOBUS_LOCATION/lib
  server     = GLOBUS_LOCATION/sbin/globus-gatekeeper
  server_args = -conf GLOBUS_LOCATION/etc/globus-gatekeeper.conf
  disable    = no
}
```

This entry has changed from the entry provided for the gatekeeper in the Globus Toolkit 2.0 Administrator's Guide. The reason is that if you followed the instructions from the install section, you do not have a static gatekeeper. This requires you to set the LD_LIBRARY_PATH so that the gatekeeper can dynamically link against the libraries in \$GLOBUS_LOCATION/lib. To accomplish the setting of the environment variable in xinetd, we use the "env =" option to set LD_LIBRARY_PATH in the gatekeeper's environment.

The advantage of this setup is that when you apply a security update to your installation, the gatekeeper will pick it up dynamically without your having to rebuild it.

After you have added the globus-gatekeeper service to either inetd or xinetd, you will need to notify inetd (or xinetd) that its configuration file has changed. To do this, follow the instructions for the server you are running below.

Inetd

On most Linux systems, you can simply run `killall -HUP inetd`. On other systems, the following has the same effect:
ps aux | grep inetd | awk '{print \$2;}' | xargs kill -HUP

Xinetd

On most linux systems, you can simply run `/etc/rc.d/init.d/xinetd restart`. Your system may also support the "reload" option. On other systems (or if that doesn't work), see man xinetd.

At this point, your gatekeeper will start up when a connection comes in to port 2119, and will keep a log of its activity in `$GLOBUS_LOCATION/var/globus-gatekeeper.log`. However, it does not yet have any authorization mapping between certificate subjects and usernames. You will need to create a file named `/etc/grid-security/grid-mapfile` which consists of single line entries listing a certificate subject and a username, like this:

```
"/O=Grid/O=Globus/OU=your.domain/CN=Your Name"    youruserid
```

You can check your subject name using `grid-cert-info -subject`. There are utility commands in `$GLOBUS_LOCATION/sbin/grid-mapfile*` for adding entries, removing entries, and checking consistency.

3. Advanced Configuration

Advanced configuration of GRAM consists of the following tasks:

1. [Adding jobmanagers](#)⁷
2. [Adding trust to a new CA/removing trust from an old CA](#)⁸
3. [Starting your own CA](#)⁹

1. Adding jobmanagers

For information about how to add a job manager for Condor, PBS, or LSF please look [here](#)¹⁰

2. Adding trust to a new CA/removing trust from an old CA

The set of trusted Certificate Authorities is contained in the `/etc/grid-security/certificates` directory. By default, that directory contains two entries. One, called `42864e48.0` is the public certificate of the Globus CA. The other, called `42864e48.signing_policy` is the signing policy for the Globus CA certificate.

The name "42864e8" comes from the `openssl -hash` option. If you create your own Certificate Authority, you can use the command `openssl x509 -in yourcert.pem -noout -hash` to determine its hash value. You will need to place a copy of that public certificate, under the name `hash.0` (where "hash" corresponds to the output of the `openssl` command) in the `/etc/grid-security/certificates` of every Toolkit installation which you want to trust certificates which your CA has signed. Additionally, you will have to create a `hash.signing_policy` file which contains the DN of your CA, as well as the namespace for which your CA signs.

Namespaces for CAs are designed to be unique. If you do establish your own CA, do not use the `"/O=Grid/O=Globus"` namespace. That is reserved for the Globus CA.

Removing trust for a particular CA is as easy as deleting the two files which correspond to the CA. First, look for the `.signing_policy` which corresponds to the CA you want to remove. Then remove both the `.signing_policy` and `.0` file that correspond to that hash.

3. Starting your own CA

There is a Globus package named [Simple CA](#) which is designed to help you establish a CA for your test Grid.

⁷ #add-jobmanagers

⁸ #add-ca

⁹ #start-ca

¹⁰ #gram2-adding-jobmanager

Chapter 3. Job Manager

The GRAM Job Manager program starts and monitors jobs on behalf of a GRAM client application. The job manager is typically started by the Gatekeeper program. It interfaces with a local scheduler to start jobs based on a job request RSL string.

1. Job Manager Setup

Job managers for Fork, PBS, LSF and Condor are included in the toolkit. But only the fork job manager is installed by default during a normal installation of the toolkit. The others must be installed separately if they are needed.

To install them from a source distribution, follow these steps:

1. go to the installer directory (e.g. gt4.2.0-all-source-installer)
2. `make gt4-gram-[pbs|lsf|condor]`
3. `make install`

Using PBS as the example, make sure the scheduler commands are in your path (qsub, qstat, pbsnodes). For PBS, another setup step is required to configure the remote shell for rsh access:

```
% cd $GLOBUS_LOCATION/setup/globus
% ./setup-globus-job-manager-pbs --remote-shell=rsh
```

The following links give extra information what parameters can be added to the setup scripts of the different scheduler adapters:

- [Condor Job Manager Setup](#)¹
- [PBS Job Manager Setup](#)²
- [LSF Job Manager Setup](#)³

2. Job Manager Configuration

[Job Manager Configuration](#)⁴

3. RSL Validation File Format

[RSL Validation File Format](#)⁵

4. Job Execution Environment

[Job Execution Environment](#)⁶

¹ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager_setup_condor/html/main.html

² http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager_setup_pbs/html/main.html

³ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager_setup_lsf/html/main.html

⁴ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager/html/globus_gram_job_manager_configuration.html

⁵ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager/html/globus_gram_job_manager_rsl_validation_file.html

⁶ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager/html/globus_gram_job_manager_job_execution_environment.html

5. RSL attributes

RSL Attributes⁷

6. Adding job managers

The fork job manager scheduler will be installed during a normal installation of the toolkit and will be installed as the default job manager service (e.g. \$GLOBUS_LOCATION/grid-services/jobmanager). Additional job manager scheduler packages installed will be installed using the convention "jobmanager-<scheduler-name>" (e.g. \$GLOBUS_LOCATION/grid-services/jobmanager-pbs).

Information on how to install an additional job manager for Condor, PBS or LSF can be found [here](#)⁸.

All job manager scheduler setup packages have the argument "-service-name <name>" in order to install a non-fork scheduler as the default job manager service. For example, this command will set the pbs scheduler as the default job manager service:

```
% setup-globus-job-manager-pbs -service-name jobmanager
```

If you need to alter the behavior of the job manager scheduler interface, or you want to create a new job manager scheduler interface for a scheduler that is not available, see this tutorial web page. The details of how to make a client submit to a non-default gatekeeper is covered in the user's guide section.

Note: If you wish to have your job manager report into your MDS, you need to install the appropriate GRAM Reporter setup package for your scheduler. The GRAM Reporter setup packages for each scheduler can be found on the [download page](#)⁹.

The details of how to make a client submit to a non-default gatekeeper is covered in the user's guide section.

Note: If you wish to have your job manager report into your MDS, you need to install the appropriate GRAM Reporter setup package for your scheduler. The GRAM Reporter setup packages for each scheduler can be found on the [download page](#)¹⁰.

⁷ http://www.globus.org/api/c-globus-4.2.0/globus_gram_job_manager/html/globus_job_manager_rsl.html

⁸ #gram2-admin-jobmanager-setup

⁹ <http://www.globus.org/toolkit/downloads/development/>

¹⁰ <http://www.globus.org/toolkit/downloads/development/>

Chapter 4. Scheduler Event Generator / Job Manager Integration

1. Introduction

This option is a method for the GRAM2 Job Manager to monitor the jobs it submits to the local scheduler. After installing, you can configure a job manager to use the new event based method for monitoring jobs, instead of the script-based polling implementation.

This change consists of a few parts

- A new script `globus-job-manager-event-generator` which translates scheduler-specific log information to a general form which the job manager can parse. This script may need to be run as a privileged account in order to parse the log files, depending on the log permissions. This script **MUST** be running in order for Job Manager processes to receive job state change notifications from the scheduler.
- A new SEG module `globus_scheduler_event_generator_job_manager` which parses a log file to determine which job state changes occur for jobs being managed by a pre-WS GRAM Job Manager.
- Changes to the `globus-gram-job-manager` program to use the Scheduler Event Generator API to look for job state change events in a log file instead using scripts to query the scheduler state.

2. `globus-job-manager-event-generator`

The `globus-job-manager-event-generator` script creates a log of all scheduler events related to a particular scheduler instance. This script was created for two purposes

- To avoid requiring that all GRAM user's have the privileges to read the scheduler's log file. Users may not be allowed read access to the scheduler's log files on all sites. The Job Manager processes is run under the user's local account (as mapped in the `gridmap` file), it is this processes that will be updated for job status via the SEG log file instead of directly from the scheduler's log file.
- To provide a simple format for the scheduler event generator logs so that the job manager will be able to quickly recover state information if the job manager is terminated and restarted. Some scheduler logs are difficult to parse, or inefficient for seeking to a particular timestamp (as is necessary for recovering job state change information). The data written by this script is easily locatably by date, and it is simple to remove old job information without compromising current job manager execution.

One instance of the `globus-job-manager-event-generator` must be running for each scheduler type to be implemented using the Scheduler Event Generator interface to receive job state changes. This program is located in the `sbin` subdirectory of the `GLOBUS_LOCATION`. The typical command line for this program is `$GLOBUS_LOCATION/sbin/globus-job-manager-event-generator -s SCHEDULER_TYPE`, where `SCHEDULER_TYPE` is the scheduler name of the Scheduler Event Generator module which should be used to generate events (`lsf`, `condor`, `pbs`).

For example, to start the event generator program to monitor an LSF batch system:

```
$GLOBUS_LOCATION/sbin/globus-job-manager-event-generator -s lsf
```

NOTE: if the `globus-job-manager-event-generator` is not running, no job state changes will be sent from any job manager program which is configured to use the Scheduler Event Generator.

3. Job Manager Configuration

By default, the job manager is configured to use the pre-WS GRAM script-based polling method. A new command line option (`-seg`) was added to the `globus-job-manager` program to enable using the Scheduler Event Generator-driven job state change notifications.

There are two ways to configure the job manager to use the scheduler event generator: globally, in the `$GLOBUS_LOCATION/etc/globus-job-manager.conf` file, or on a per-service basis in the service entry file in the `$GLOBUS_LOCATION/etc/grid-services` directory.

3.1. Global Job Manager Configuration

To enable using the Scheduler Event Generator interface for all Job Managers started from a particular `GLOBUS_LOCATION`, add a line containing the string

```
-seg
```

to the file `$GLOBUS_LOCATION/etc/globus-job-manager.conf`.

EXAMPLE `$GLOBUS_LOCATION/etc/globus-job-manager.conf`:

```
-home "/opt/globus"
-globus-gatekeeper-host globus.yourdomain.org
-globus-gatekeeper-port 2119
-globus-gatekeeper-subject "/O=Grid/OU=Your Organization/CN=host/globus.yourdomain.org"
-globus-host-cputype i686
-globus-host-manufacturer pc
-globus-host-osname Linux
-globus-host-osversion 2.6.10
-save-logfile on_error
-state-file-dir /opt/globus/tmp/gram_job_state
-machine-type unknown
-seg
```

3.2. Scheduler-specific Job Manager Configuration

To enable using the Scheduler Event Generator interface for a particular Job Manager, add the string `-seg` to the end of the line in the service's file in the `$GLOBUS_LOCATION/etc/grid-services` directory.

EXAMPLE `$GLOBUS_LOCATION/etc/grid-services/jobmanager-lsf`:

```
stderr_log,local_cred - /opt/globus/libexec/globus-job-manager globus-job-manager -conf /o
```

No SEG with Job Manager fork

The Job Manager fork does not support using the Scheduler Event Generator. If the `-seg` option is passed to a fork Job Manager, it will be ignored.

4. globus-job-manager-event-generator Configuration

The globus-job-manager-event-generator program requires that the globus_job_manager_event_generator setup package be installed and run. This setup package creates the `$GLOBUS_LOCATION/etc/globus-job-manager-seg.conf` file and initializes a directory to use for the scheduler logs.

By default, this setup script will create a configuration entry and directory for each scheduler installed on the system. For each scheduler to be handled by the globus-job-manager-event-generator program, there must be an entry in the file in the pattern:

```
<SCHEDULER_TYPE>_log_path=<PATH>
```

The two variable substitutions for this pattern are

SCHEDULER_TYPE

Must match the name of the scheduler-event-generator module for the scheduler (supported with GT 4.2 are lsf, condor, and pbs).

PATH

A path to a directory which must be writable by the account which will run the globus-job-manager-event-generator program for the SCHEDULER_TYPE, and world-readable (or readable for a group which contains all users which will run jobs via GRAM on that system). Each directory specified in the configuration file must be unique, or behavior is undefined.

EXAMPLE `$GLOBUS_LOCATION/etc/globus-job-manger-seg.conf`:

```
lsf_log_path=/opt/globus/var/globus-job-manager-seg-lsf
pbs_log_path=/opt/globus/var/globus-job-manager-seg-pbs
```

In this example, pbs and lsf schedulers are configured to use distinct subdirectories of the `/opt/globus/var/` directory.

NOTE: For best performance, the log paths should be persistent across system reboots and mounted locally (non-networked).

NOTE: If a scheduler is added after the configuration step is done, administrator must rerun the setup package's script (`$GLOBUS_LOCATION/setup/globus/setup-seg-job-manager.pl`) or modify the configuration file and create the required directory with appropriate permissions.

5. Running the globus-job-manager-event-generator

The globus-job-manager-event-generator must be running when jobs are submitted to the Job Manager if job state changes are to be detected. One instance of the globus-job-manager-event-generator program must be running for each scheduler type which is handled by a Job Manager and configured to use the Scheduler Event Generator interface.

The command line for the `globus-job-manager-event-generator` program is `globus-job-manager-event-generator -s SCHEDULER_TYPE`. The `SCHEDULER_TYPE` should match the pattern of a `log_path` entry in the `$GLOBUS_LOCATION/etc/globus-job-manager-seg.conf` as described above.

NOTE: Remember, if your scheduler logs have restrictive permissions, then this script must be run by an account which has privileges to read those files.

NOTE: Old log files created by the `globus-job-manager-event-generator` script may be deleted if the administrator is certain that there are no jobs which will restart and require the old information. The names of the log files correspond to the dates when the events occurred. If there is at least one log file in the directory, then when the `globus-job-manager-event-generator` is restarted, it will resume logging from the timestamp of the newest event in that log file.

6. Troubleshooting the `globus-job-manager-event-generator`

PROBLEM: The `globus-job-manager-event-generator` program terminates immediately with the output:

```
Error: SCHEDULER not configured
```

SOLUTION 1: Make sure that you specified the correct name for the `SCHEDULER` module on the command line to the `globus-job-manager-event-generator` program

SOLUTION 2: There is no entry for `lsf` in the `$GLOBUS_LOCATION/etc/globus-job-manager-seg.conf` file. See the section on `globus-job-manager-event-generator` Configuration.

PROBLEM: The `globus-job-manager-event-generator` program terminates immediately with the output:

```
Fault: globus_xio: Operation was canceled
```

SOLUTION: The scheduler module selected on the command line could not be loaded by the Globus Scheduler Event Generator. Check that the name is correct, the module is installed, and the setup script for that module has been run.

PROBLEM: The Job Manager never receives any events from the scheduler.

SOLUTION 1: Verify that the directory specified in the `$GLOBUS_LOCATION/etc/globus-job-manager-seg.conf` for the scheduler exists, is writable by the account running the `globus-job-manager-event-generator` and is readable by the user account running the job manager.

SOLUTION 2: Verify that the `globus-job-manager-event-generator` program is running.

SOLUTION 3: Verify that the `globus-job-manager-event-generator` program has permissions to read the scheduler logs. To help diagnose this, run (as the account you wish to run the `globus-job-manager-event-generator` as) the command

```
$GLOBUS_LOCATION/libexec/globus-scheduler-event-generator -s <SCHEDULER_TYPE> -t 1
```

You should see events printed to the stdout of that process if it is working correctly.

Chapter 5. Audit Logging



Note

For more information, click [here](#).

DRAFT

Chapter 6. Testing GRAM

First launch a gatekeeper by running the following (as yourself, not root):

```
% grid-proxy-init -debug -verify
  % globus-personal-gatekeeper -start
```

This command will output a contact string like `hostname:4589:/O=Grid/O=Globus/CN=Your Name`. Substitute that contact string for `<contact>` in the following command:

```
% globus-job-run <contact> /bin/date
```

You should see the current date and time. At this point you can stop the personal gatekeeper and destroy your proxy with:

```
% globus-personal-gatekeeper -killall
  % grid-proxy-destroy
```

Please note that the above instructions are just for testing, and do not install a fully functioning gatekeeper on your machine for everyone to use. Installing a system-level gatekeeper for everyone to use will be covered in the [configuration section](#)¹ of this guide.

¹ [#gram2-admin-configuring](#)

Chapter 7. Usage statistics collection by the Globus Alliance

No usage statistic package is sent after the completion of a job like it's done in WS-GRAM (see [here](#)¹).

DRAFT

¹ [../wsgram/admin-index.html#s-wsgram-admin-usage](#)